

PERENCANAAN DAN PEMBUATAN SISTEM AKUSISI DATA BERBASIS IBM-PC

TUGAS AKHIR



RSE
621.398.14
Heri
D-1
1998

| | |
|---------------------|-------------|
| PERPUSTAKAAN ITS | |
| Tgl. Terima | 23 - 6 - 00 |
| Tgl. Pengemb. | 17 |
| No. Revisi | 2/231 |

Oleh :

RASTONO HENDRIYANTO

NRP. 2295 030 025

MOCHAMAD ISLAN

NRP. 2295 030 045

PROGRAM STUDI DIPLOMA III KOMPUTER KONTROL
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
1998



PERENCANAAN DAN PEMBUATAN SISTEM AKUSISI DATA BERBASIS IBM-PC

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Menyelesaikan Studi
Program Diploma III Komputer Kontrol
Pada
Jurusan Teknik Elektro
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
S u r a b a y a**

**Mengetahui / Menyetujui
Dosen Pembimbing**



**Ir. HENDRA KUSUMA
NIP. 131 846 104**

**SURABAYA
NOPEMBER, 1998**

ABSTRAK

Pengukuran sinyal-sinyal yang berbentuk analog sering dilakukan dalam berbagai bidang kehidupan. Dalam pengukuran sinyal-sinyal tersebut dibutuhkan suatu alat ukur yang memiliki tingkat presisi yang tinggi. Dengan perkembangan teknologi, diciptakan suatu peralatan yang dapat melakukan pengukuran data yang hasilnya diolah dan ditampilkan dalam komputer. Dalam Tugas Akhir ini akan dibahas peralatan ukur sinyal analog dengan metode Analog to Digital Converter yang diinterfacekan pada IBM-PC.

Sistem akusisi data yang dibahas tersebut memiliki beberapa chanel input yang masing-masing chanel dapat berisi sinyal dari sensor-sensor dengan jangkauan ukur tiap-tiap sensor 0 - 2V. Dengan resolusi pembacaan data Analog to Digital Converter 1mV maka sistem akusisi data tersebut memiliki tingkat ketelitian yang tinggi.

Pada tugas akhir ini ada 8 chanel input yang dapat diukur, namun hanya 3 chanel input yang diaplikasikan dalam bentuk sinyal tegangan dari sensor-sensor suhu dan rangkaian pembagi tegangan. Sinyal-sinyal tersebut diolah dan ditampilkan dalam bentuk tabel numerik oleh IBM-PC, sehingga dapat dianalisa dengan mudah dan efisien.

KATA PENGANTAR

Alhamdulillah, kami panjatkan puji syukur kehadiran Allah S.W.T yang karena rahmat, hidayah serta syafaat-Nya kami dapat menyelesaikan penyusunan tugas akhir yang berjudul :

"PERENCANAAN DAN PEMBUATAN SISTEM AKUSISI DATA BERBASIS IBM-PC"

Tugas akhir ini disusun untuk memenuhi persyaratan yang telah ditentukan untuk menyelesaikan kuliah di Program Studi D3 Komputer Kontrol Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya.

Penyelesaian tugas akhir yang memiliki bobot 4 SKS ini berdasarkan teori-teori yang telah kami peroleh selama kuliah, literatur-literatur, bimbingan dari dosen, serta pihak-pihak yang telah membantu dalam penyusunannya.

Kami sadari dalam penyusunan tugas akhir ini tak luput dari kekurangan oleh karena itu kami mengharapkan saran dan kritik demi kesempurnaan tugas akhir ini. Besar harapan kami, semoga buku ini dapat bermanfaat bagi yang

memerlukannya dan bahkan dapat dikembangkan lebih lanjut sehingga mempunyai manfaat dan daya guna yang lebih baik bagi perkembangan ilmu pengetahuan dan teknologi serta kehidupan masyarakat.

Surabaya, Nopember 1998

Penyusun

UCAPAN TERIMA KASIH

Dengan telah tersusunnya buku Tugas Akhir ini kami, Hendri selaku salah satu penyusun buku mengucapkan banyak terima kasih kepada :

1. Bapak dan Ibu tercinta yang telah banyak memberikan saran, dorongan dan bantuan dalam bentuk moril maupun materiil kepada kami.
2. Ir. HM. Djoko Santoso, selaku Ketua Program Studi D3 Komputer Kontrol FTI - ITS.
3. Ir. Hendra Kusuma, selaku dosen pembimbing yang telah berkenan membimbing dan membantu kami dalam menyelesaikan tugas akhir ini.
4. Ir. Henny Utami, selaku dosen wali yang telah memberikan bimbingan dan bantuan dalam menyelesaikan tugas akhir ini.
5. Ir. Rusdhianto, MT, selaku dosen D3 Komputer Kontrol yang telah berkenan membantu kami dalam menyelesaikan tugas akhir ini.
6. Mbak Sita, dan adik-adikku (Rahma dan Tini) yang telah membantu dan memberikan dorongan semangat dalam penyelesaian tugas akhir ini.
7. Rekan TA' ku Islan, dan sahabat-sahabatku Ifa, Mungki, Bakar, Hermain, Agung, Semin dan rekan-rekan Computer Control angkatan '95 yang telah membantu kami dalam menyelesaikan tugas akhir ini.

UCAPAN TERIMA KASIH

Dengan telah tersusunnya buku Tugas Akhir ini kami, Islan selaku salah satu penyusun buku mengucapkan banyak terima kasih kepada :

1. Bapak dan Ibu yang secara tidak langsung telah membantu dalam penyelesaian tugas akhir ini.
2. Ir. HM. Djoko Santoso, selaku Ketua Program Studi D3 Komputer Kontrol FTI - ITS.
3. Ir. Hendra Kusuma, selaku dosen pembimbing yang telah berkenan membimbing dan membantu kami dalam menyelesaikan tugas akhir ini.
4. Ir. Arif Musthofa, selaku dosen wali yang telah memberikan bimbingan dan bantuan dalam menyelesaikan tugas akhir ini.
5. Rekan TA' ku Hendri dan rekan-rekan Computer Control angkatan '95 yang telah berkenan membantu kami dalam menyelesaikan tugas akhir ini.
6. Imel yang setia memberikan dorongan dalam penyelesaian tugas akhir ini.
7. "STUDIO" Sablon, Cacahoo dan Wira beserta semua pihak yang telah membantu terwujudnya buku tugas akhir ini.

DAFTAR ISI

| | Halaman |
|--|---------|
| HALAMAN JUDUL | i |
| HALAMAN PENGESAHAN | ii |
| ABSTRAK | iii |
| KATA PENGANTAR | iv |
| UCAPAN TERIMA KASIH | vi |
| DAFTAR ISI | viii |
| DAFTAR GAMBAR | xi |
| DAFTAR TABEL | xii |
| | |
| BAB I. PENDAHULUAN | 1 |
| 1.1. Latar Belakang Masalah | 1 |
| 1.2. Maksud Dan Tujuan | 2 |
| 1.3. Permasalahan | 2 |
| 1.4. Pembatasan Masalah | 3 |
| 1.5. Sistematika Pembahasan | 4 |
| | |
| BAB II. TEORI PENUNJANG | 6 |
| 2.1. Teori Akusisi Data | 6 |
| 2.2. Sistem I/O Pada IBM PC | 7 |
| 2.2.1. Slot Ekspansi Pada IBM PC | 7 |

| | |
|---|-----------|
| 2.2.2. Siklus Baca Dari I/O Port. | 14 |
| 2.2.3. Siklus Tulis Dari I/O Port. | 15 |
| 2.2.4. Pemetaan I/O Port IBM PC. | 16 |
| 2.2.5. Address Decoder. | 18 |
| 2.3. Programmable Peripheral Interface 8255. | 19 |
| 2.3.1. Fungsi Pin-Pin Pada PPI 8255. | 20 |
| 2.3.2. Mode Kerja PPI 8255. | 22 |
| 2.4. Konversi Data Analog Ke Digital. | 24 |
| 2.4.1. Analog To Digital Converter 0804. | 25 |
| 2.5. Sensor Suhu. | 26 |
| 2.5.1. IC LM 35. | 28 |
| 2.6. Rangkaian Penguat Operasional. | 29 |
| 2.6.1. Penguat Operasional Sebagai Voltage Follower. | 30 |
| 2.6.2. Penguat Operasional Sebagai Penguat Pembalik. | 31 |
| 2.6.3. Penguat Operasional Sebagai Penguat Tak Membalik. | 32 |
| 2.7. Rangkaian Filter Pasif. | 33 |
| BAB III PERENCANAAN DAN PEMBUATAN PERANGKAT KERAS | 34 |
| 3.1. Blok Diagram Rangkaian Akusisi Data. | 34 |
| 3.2. Prinsip Kerja Rangkaian Akusisi Data. | 35 |
| 3.3. Rangkaian Interface PPI 8255. | 36 |
| 3.4. Rangkaian Konversi Data Analog ke Digital. | 39 |
| 3.5. Rangkaian Pembangkit Tegangan Referensi. | 40 |

| | |
|--|-------------|
| 3.6. Rangkaian Pengatur Jangkauan Ukur | 42 |
| 3.7. Rangkaian Pengatur Chanel Input | 45 |
| 3.8. Rangkaian Pengukur Suhu | 47 |
| 3.8.1. Rangkaian Sensor Suhu IC LM 35 | 47 |
| 3.9. Rangkaian Pembagi Tegangan Sebagai Input Chanel | 48 |
| BAB IV PERENCANAAN PERANGKAT LUNAK | 50 |
| BAB V PENGUKURAN DAN PENGUJIAN PERALATAN | 55 |
| 5.1. Pengukuran Rangkaian Input ADC | 55 |
| 5.1.1. Rangkaian Sensor Suhu IC LM 35 | 55 |
| 5.1.2. Rangkaian Pembagi Tegangan | 57 |
| 5.2. Pengujian PPI 8255 | 58 |
| 5.3. Pengujian ADC 0804 | 61 |
| BAB VI PENUTUP | 63 |
| 6.1. Kesimpulan | 63 |
| 6.2. Saran | 64 |
| DAFTAR PUSTAKA | xii |
| LAMPIRAN | xiii |
| USULAN TUGAS AKHIR | xiv |
| RIWAYAT HIDUP | xv |

DAFTAR GAMBAR

| | Halaman |
|---|---------|
| Gambar 2.1 Slot Ekspansi pada IBM PC | 14 |
| Gambar 2.2 Siklus Baca dari I/O Port | 15 |
| Gambar 2.3 Siklus Tulis dari I/O Port | 16 |
| Gambar 2.4 Konfigurasi Address Untuk I/O Port | 18 |
| Gambar 2.5 Blok Diagram PPI 8255 | 20 |
| Gambar 2.6 PPI 8255 | 22 |
| Gambar 2.7 Format Control Word Register PPI 8255 | 24 |
| Gambar 2.8 Susunan Pin ADC 0804 | 26 |
| Gambar 2.9 IC LM 35 | 29 |
| Gambar 2.10 Voltage Follower | 31 |
| Gambar 2.11 Rangkaian Penguat Pembalik | 32 |
| Gambar 2.12 Rangkaian Penguat Tak Membalik | 32 |
| Gambar 2.13 Rangkaian Low Pass Filter | 33 |
| Gambar 3.1 Blok Diagram Keseluruhan Rangkaian Akusisi Data | 34 |
| Gambar 3.2 Blok Diagram Sistematika Kerja Rangkaian Akusisi Data | 35 |
| Gambar 3.3 Rangkaian PPI 8255 | 38 |
| Gambar 3.4 Rangkaian ADC | 40 |
| Gambar 3.5 Rangkaian Tegangan Referensi | 42 |
| Gambar 3.6 Rangkaian Jangkauan Ukur | 43 |
| Gambar 3.7 Rangkaian Chanel Input | 46 |
| Gambar 3.8 Rangkaian Sensor IC LM35 | 48 |
| Gambar 3.9 Rangkaian Pembagi Tegangan | 49 |
| Gambar 4.1 Flowchart Pemrograman Rangkaian Akusisi Data | 54 |
| Gambar 5.1 Rangkaian Pengukuran Sensor Suhu IC LM35 | 56 |
| Gambar 5.2 Rangkaian Pengukuran Pembagi (Tegangan Potensiometer). | 57 |
| Gambar 5.3 Rangkaian Pengujian PPI 8255 | 59 |
| Gambar 5.4 Rangkaian Pengujian ADC | 62 |

DAFTAR TABEL

| | Halaman |
|--|---------|
| Tabel 2.1 Pemetaan Alamat Port I/O IBM PC | 17 |
| Tabel 3.1 Tabel Address Decoder Untuk Alamat 300H-303H | 36 |
| Tabel 3.2 Penentuan Alamat Pada Input-Output Decoder | 37 |
| Tabel 5.1 Hasil Pengukuran Tegangan Output Sensor Suhu IC LM35 | 56 |
| Tabel 5.2 Hasil Pengukuran Tegangan Output Rangkaian Pembagi Tegangan (Potensiometer) | 57 |
| Tabel 5.3 Hasil Pengukuran Tegangan Output PPI 8255 | 60 |
| Tabel 5.4 Hasil Pengujian ADC | 61 |

BAB I

PENDAHULUAN

1.1. Latar Belakang

Dalam kehidupan sehari-hari manusia seringkali dihadapkan pada serangkaian masalah yang memerlukan solusi yang tepat. Adakalanya solusi tersebut berupa alat bantu untuk memudahkan manusia dalam mengerjakan suatu tugas. Penggunaan elektronika sebagai alat bantu manusia sudah tidak diragukan lagi, termasuk dalam hal pengukuran. Dengan berkembangnya teknologi, tuntutan akan kebutuhan alat ukur yang terpercaya dan lebih teliti semakin meningkat yang kemudian menghasilkan perkembangan-perkembangan baru dalam perencanaan dan pemakaiannya.

Pada era komputerisasi ini pengukuran dan pembacaan data analog dapat memiliki ketelitian dan akurasi yang tinggi dengan pemanfaatan komputer. Oleh karena itu dibutuhkan suatu alat berupa ADC untuk merubah besaran atau variabel ukur menjadi sinyal digital agar dapat dibaca oleh komputer untuk kemudian ditampilkan ke layar monitor. Pada perkembangan selanjutnya data pengukuran tersebut dapat dikirimkan pada komputer lainnya sehingga membentuk sistem akuisisi data.

1.2. Maksud dan Tujuan

Maksud dari penyusunan tugas akhir ini adalah mengukur dan membaca data yang berbentuk sinyal analog yang diperoleh dari sensor atau transduser untuk kemudian ditampilkan ke layar monitor.

Dalam tugas akhir ini tujuan yang dikehendaki adalah untuk membuat suatu sistem akuisisi data dimana sistem ini berfungsi mengukur dan membaca data analog dari sejumlah sensor dan diinterfacekan dengan IBM-PC sehingga data tersebut dapat dianalisa dan ditampilkan dalam layar monitor. Dalam hal pengukuran data ini diusahakan rangkaian yang didukung ADC tersebut dapat memiliki ketelitian yang tinggi sehingga sinyal yang dihasilkan keluaran sesuai dengan sinyal masukan yang berbentuk analog.

1.3. Permasalahan

Pengukuran parameter data misalnya suhu, umumnya dilakukan dengan penggunaan sensor atau transduser yang langsung dibaca dengan tenaga manusia. Pembacaan data yang dilakukan dirasa kurang efisien selain itu kemungkinan terjadinya kesalahan terutama dari segi ketelitian pengukuran.

Permasalahan yang diangkat dalam tugas akhir ini adalah bagaimana cara memanfaatkan komputer untuk digunakan dalam

pembacaan data dari berbagai macam sensor analog yang letaknya tidak selalu pada lokasi yang sama dengan tanpa mengurangi tingkat ketelitian dan akurasi yang tinggi. Tentu saja dalam pengiriman parameter data ini membutuhkan rangkaian eksternal komputer dan rangkaian interface. Pada tugas akhir ini letak sensor yang digunakan tidak jauh dari rangkaian eksternal atau ADC sehingga hanya diperlukan kabel penghubung sensor. Pada sistem akuisisi data ini sensor yang dapat diukur tidak hanya sensor suhu namun dapat berupa berbagai sensor. Hanya saja dalam tugas akhir ini sensor yang digunakan adalah sensor suhu.

1.4. Pembatasan Masalah

Dalam tugas akhir sistem akuisisi data ini dibatasi oleh beberapa pokok bahasan yaitu :

1. Pada sistem akuisisi data ini input yang digunakan ada 2 buah yaitu sensor suhu LM35 dan rangkaian pembagi tegangan Potensiometer yang tegangan outputnya hanya sampai 2 Volt.
2. Data pengukuran sinyal analog input sistem yang diperoleh dari sensor berupa sinyal tegangan DC.
3. Channel input yang dapat digunakan dalam sistem ini ada 8 channel dengan pengaturan channel melalui software. Dari 8 channel input yang ada hanya 2 channel yang digunakan.

4. Proses sampling data dalam tugas akhir ini menggunakan software.
5. Pada sistem ini jangkauan ukur dari masing-masing sinyal input analog antara 0 - 2040 mV dengan resolusi pengukuran sebesar 1mV.
6. Software dalam sistem akusisi data ini digunakan untuk membaca dan menampilkan data dari sinyal input.

1.5. Sistematika Pembahasan

Pada tugas akhir ini sistematika pembahasannya adalah sebagai berikut :

Bab I : Pendahuluan

Pada bab ini membahas tentang latar belakang, maksud dan tujuan, pembatasan masalah dan sistematika pembahasan.

Bab II : Teori Penunjang

Pada bab ini membahas tentang teori-teori penunjang yang menjelaskan sistem akusisi data, sistem operasi I/O dari IBM PC, teknik interface pada IBM PC konversi data dari analog ke digital, sensor-sensor suhu dan kinerja rangkaian penguat operasional.

Bab III : Perencanaan dan Pembuatan Perangkat Keras

Pada bagian ini membahas tentang perencanaan perangkat keras dan analisa rangkaian-rangkaiannya di dalam sistem akusisi data.

Bab IV : Perencanaan Perangkat Lunak

Pada bagian ini membahas tentang perencanaan perangkat lunak untuk mengaktifkan sistem akusisi data ini.

Bab V : Pengukuran dan Pengujian Alat

Pada bab ini membahas tentang pengukuran dan pengujian alat beserta analisisnya.

Bab VI : Penutup

Pada bab ini menjelaskan tentang kesimpulan dari tugas akhir ini dan saran-saran untuk pengembangan sistem akusisi data tersebut.

BAB II

TEORI PENUNJANG

2.1. Teori Akusisi Data

Pengambilan data dari sejumlah terminal atau modul sensor yang letaknya berjauhan dari komputer pusat dapat dilakukan dengan cara langsung yaitu melalui hubungan antarmuka atau interface yang dapat berupa kabel RS-232C, modem dan radio serta dapat pula melalui cara paralel dengan PPI 8255. Sistem pengiriman data ini disebut Data Acquisition atau Akusisi Data. Data yang dikirimkan oleh terminal atau modul sensor umumnya berbentuk sinyal analog, sedangkan data yang diterima komputer adalah berbentuk digital sehingga data yang dikirimkan sensor tersebut harus dikonversikan dahulu ke dalam bentuk digital dengan sebuah peralatan yaitu ADC. Dalam sistem akusisi data, hal yang ditekankan adalah tingkat ketelitian yang tinggi untuk pembacaan dan ketepatan dalam pengambilan data. Dengan adanya kedua hal tersebut maka data digital yang diterima komputer dapat benar-benar sesuai dengan sinyal input analognya. Salah satu sifat dari sistem akusisi data ini adalah sistem hanya dapat mengambil data dari terminal dikirimkan ke komputer pusat, sedangkan data yang diproses dalam komputer tidak dapat dikirimkan kembali ke terminal.

2.2. Sistem I/O Pada IBM PC

Dalam proses pengoperasiannya IBM PC dilengkapi dengan sistem I/O yang berfungsi menghubungkan processor dengan peralatan luar. Sistem ini dibentuk untuk memisahkan proses pembacaan/penulisan input dan output dengan pembacaan/penulisan memori. Sistem ini mencakup lima hal yang akan dijelaskan masing-masing yaitu slot ekspansi IBM PC, siklus baca dari I/O port, siklus tulis dari I/O port, pemetaan I/O dan address decoder untuk pengalamatan masing-masing I/O.

2.2.1. Slot Ekspansi Pada IBM PC

Dalam IBM PC terdapat beberapa slot ekspansi yang terhubung secara paralel dan mempunyai konfigurasi pin yang sama dan dapat mengakses memori dan I/O map yang sama. Masing-masing slot mempunyai 62 pin pada kedua sisinya. Pada slot-slot inilah peralatan interface diletakkan. Isi masing-masing slot secara umum dapat dibagi menjadi 8 bagian yaitu :

1. 8 bit sistem data bus.
2. 20 bit address bus.
3. Sumber daya dan ground untuk rangkaian interface.
4. Clock dan timing signal.
5. Jalur-jalur control untuk memori I/O read atau I/O write.
6. Jalur maskable dan non maskable Interrupt request.
7. Jalur status untuk Interrupt enable flag.
8. Jalur status bus cycle.

Gambar dibawah ini menunjukkan konfigurasi pin-pin yang tersedia pada slot IBM PC. Dari 62 pin, hanya beberapa pin yang digunakan untuk keperluan interfacing. Fungsi pin-pin pada slot ekspansi ini dapat dijelaskan sebagai berikut :

- OSC : Pin ini merupakan pin output yang mengeluarkan sinyal dengan frekwensi 198 MHz dan siklus kerjanya 50%. Semua sinyal timing berasal dari sinyal ini.
- CLOCK : Sinyal ini berasal dari sinyal OSC yang telah diterangkan diatas. Sinyal ini diperoleh dari pembagian dengan faktor 3 sehingga memberikan frekwensi 66 MHz. Siklus kerja sinyal ini adalah 33,3% dan periodenya 15ns, sehingga waktu high adalah 5ns dan waktu low 10ns.
- RESET : Sinyal ini merupakan sinyal output. Pin ini menjadi berlogika tinggi selama sistem dinyalakan, dan pada saat sistem di-reset. Sinyal ini diatur dari IC 8284A.
- ALE : Sinyal ini merupakan sinyal output yang dikemudikan dari IC bus controller, sinyal ini digunakan untuk mengindikasikan bahwa address bus sedang valid selama satu siklus bus. Pin ini berlogika tinggi selama address bus valid. Sinyal ini digunakan untuk me-latch address dari mikroprosesor, karena dalam arsitektur ada pin-pin address yang dimultiplex dengan sinyal lain. Pin ALE ini merupakan pin yang berguna untuk

melihat siklus-siklus yang dilakukan oleh prosesor, karena sinyal ALE merupakan awal dari setiap siklus bus. Selama siklus DMA, sinyal ALE menjadi non aktif.

IOR : Pin ini merupakan pin output yang dikemudikan dari IC 8288 bus controller. sinyal ini digunakan untuk mengindikasikan I/O port bahwa mikroprosesor menginisialisasi siklus baca I/O port, dan address yang ada pada address bus adalah address I/O port. Sinyal ini adalah sinyal aktif low, dan I/O port harus menempatkan datanya sekitar 30 ns sebelum rising edge dari IOR, agar prosesor memperoleh data yang valid.

IOW : Pin ini merupakan pin output yang aktif low. Pin ini dikemudikan dari IC bus controller 8288 selama siklus bus dari mikroprosesor, dan mengindikasikan bahwa address bus merupakan address I/O port dan data bus mengandung data yang akan ditulis ke I/O port.

MEMW : Pin ini merupakan pin output yang aktif low, berguna untuk menulis data dari sistem bus ke memori. Pin ini dikemudikan dari IC bus controller 8288 selama siklus bus dari prosesor, dan mengindikasikan bahwa bus address memuat address memori kemana data bus akan dituliskan.

MEMR : Pin ini merupakan pin output yang aktif low, berguna untuk membaca data dari memori ke sistem bus. Pin ini

dikemudikan dari IC bus controller 8288 selama siklus bus dari prosesor, dan mengindikasikan bahwa bus address valid dari memori yang akan dibaca isinya.

$A_{0...19}$: Pin-pin ini merupakan pin-pin address A_0 sampai A_{19} yang digunakan untuk mengakses memori pada sistem bus dan I/O. A_0 adalah Least Significant Bit (LSB) dan A_{19} adalah Most Significant Bit (MSB). Dengan instruksi IN dan OUT prosesor dapat mengakses 64K byte address I/O port. Range ini dibawa oleh jalur $A_0 - A_{15}$, sedangkan $A_{16} - A_{19}$ tidak digunakan dan dibuat nonaktif selama siklus I/O. Namun pada aplikasi PC hanya address $A_0 - A_9$ saja yang digunakan untuk meng-address I/O port. Selain itu hanya address I/O port 0200H sampai 03FFH saja yang valid pada sistem bus.

$D_{0...7}$: Kedelapan jalur ini merupakan jalur data dua arah dan digunakan untuk mengkomunikasikan data antara mikroprosessor, memori dan I/O. D_0 adalah Least Significant Bit (LSB) dan D_7 adalah Most Significant Bit (MSB).

$DRQ_{1...3}$: Pin ini merupakan pin input yang aktif high dan digunakan interface untuk meminta siklus DMA. ROM BIOS dari PC menginisialisasi DMA controller sehingga DRQ_1 mempunyai prioritas tertinggi dan DRQ_3 mempunyai prioritas terendah. DRQ_0 sebenarnya mempunyai prioritas tertinggi, tetapi ini

tidak tersedia pada sistem bus. DRQ_0 digunakan prosesor untuk melakukan penyegaran RAM dinamis.

$DACK_{0..3}$: Pin ini merupakan pin output yang aktif low yang dihasilkan oleh DMA controller 8237-5 untuk mengindikasikan bahwa sinyal DRQ yang bersesuaian telah diterima dan DMA controller akan mengambil alih bus dan melanjutkan dengan siklus DMA. Perlu dicatat bahwa tidak terdapat DRQ_0 pada sistem bus (yang bersesuaian dengan $DACK_0$). $DACK_0$ dikirimkan hanya untuk mengindikasikan bahwa siklus DMA yang ada adalah siklus dummy-read yang digunakan untuk menyegarkan memori dinamis pada sistem. Selama siklus dummy-read ini, address bus mengandung address RAM yang valid. $DACK_0$ ini muncul setiap 72 siklus clock atau $15,12\mu s$.

$IRQ_{2..7}$: Pin ini digunakan untuk menghasilkan interrupt request pada mikroprosesor dari sistem bus. Sinyal-sinyal ini dihubungkan langsung dengan IC 8259A pada sistem board. Program BIOS pada ROM menginisialisasi 8259A controller agar IRQ_2 mempunyai prioritas tertinggi dan IRQ_7 menjadi yang terendah.

I/O CHECK : Pin ini merupakan pin input yang aktif low dan digunakan untuk mendeteksi adanya kesalahan pada interface yang dipasang pada I/O port.

I/O CH RDY: Pin ini merupakan pin input yang digunakan untuk memperpanjang siklus bus, agar memori atau I/O port yang relatif lambat dari siklus bus normal (840ns) dapat dihubungkan dengan sistem. Bila akan memperpanjang siklus bus, memori atau I/O port tersebut harus membuat pin I/O port tersebut harus membuat pin I/O CH RDY menjadi rendah saat decoding dilakukan dan menerima sinyal MEMR, MEMW, IOR, IOW.

AEN : Pin ini merupakan pin output yang aktif high yang dihasilkan oleh DMA controller yang menunjukkan bahwa siklus DMA sedang dilakukan. Sinyal ini memutuskan hubungan bus lokal mikroprosesor terhadap sistem bus, serta menghubungkan sistem bus ke DMA controller. Pada sistem bus, sinyal ini berguna untuk memutuskan hubungan address I/O port selama terjadi siklus DMA ini address memori tidak di-decode oleh decoder pada I/O port. Bila ini tidak dilakukan, maka mungkin terjadi kekacauan address, karena pada saat yang sama IOW dan IOR mungkin aktif low dan address memori ada pada bus address.

TC : Pin ini merupakan pin output yang aktif high yang dihasilkan oleh DMA controller 8237-5. Sinyal ini mengindikasikan bahwa channel telah mencapai jumlah siklus transfer seperti

yang diprogramkan. Sinyal ini umumnya digunakan untuk menghentikan suatu transfer data block DMA. Sinyal T_C muncul setiap 990,804 ms yang merupakan waktu untuk siklus penyegaran 65,536 address memori yang pertama.

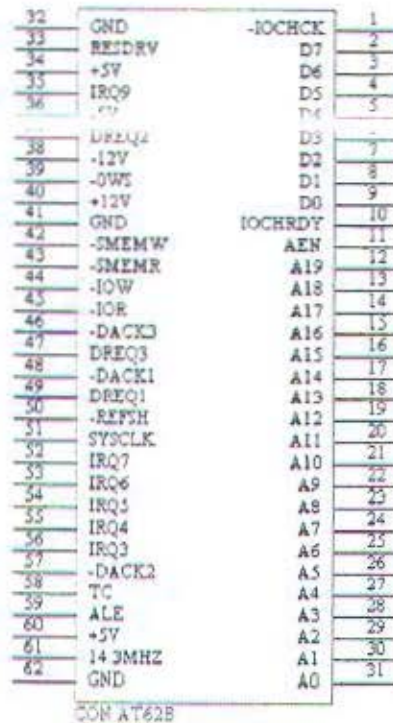
+5V DC : Tegangan 5v DC tersedia dua pin. Tegangan ada dalam batas-batas 4,75v dan 5,25v (toleransi 5%)

-5V DC : Tegangan 5v DC tersedia satu pin. Tegangan ada dalam batas-batas -4,75v dan -5,25v (toleransi 5%)

+12V DC : Tegangan 12v DC tersedia satu pin. Tegangan ada dalam batas-batas 10,8v dan 13,2v (toleransi 10%)

-12V DC : Tegangan 12v DC tersedia satu pin. Tegangan ada dalam batas-batas -10,8v dan -13,2v (toleransi 10%)

GND : Ground sistim DC dan ground frame tersedia pada tiga pin. Jalur ini merupakan jalur ground dari semua peralatan yang ada pada sistim.



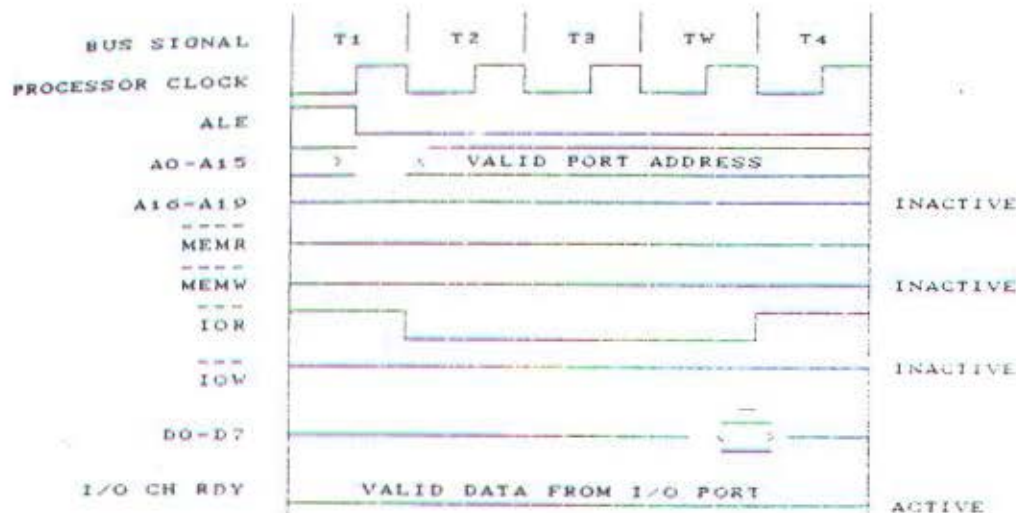
Gambar 2.1. Slot Ekspansi pada IBM PC ¹⁾

2.2.2. Siklus Baca dari I/O Port

Siklus ini aktif pada saat prosessor mengirimkan instruksi "IN". Siklus ini dipakai untuk mengambil data dari data bus. Waktu yang diperlukan dalam operasi "IN" ini sekitar 5 clock ($5 \times 15 \text{ ns} = 75 \text{ ns}$). Sebab IBM PC dapat dioperasikan pada 66 MHz. Seperti yang ditunjukkan pada gambar 2.2. adalah siklus serta jumlah waktu untuk operasi baca dari I/O port. Selama operasi baca prosessor mengirimkan 16 bit alamat ($A_0 - A_{15}$), sedangkan $A_{16} - A_{19}$ tidak diaktifkan. Dari gambar dapat dijelaskan sebagai berikut :

¹⁾ Eggebrecht, Lewis C, Interfacing to IBM-PC, Howard W. Sams & CO., Inc, Indiana Polis

Untuk siklus waktu T1, ALE diaktifkan menunjukkan bahwa address A_0-A_{15} adalah benar / valid. Pada saat siklus waktu T2, IOR diaktifkan dan port siap mengirimkan data ke data bus. Pada siklus waktu T4, Data diambil dari data bus oleh prosesor dan IOR dimatikan. Seperti pada gambar dibawah ditunjukkan wait state T_w disisipkan secara otomatis oleh prosesor.



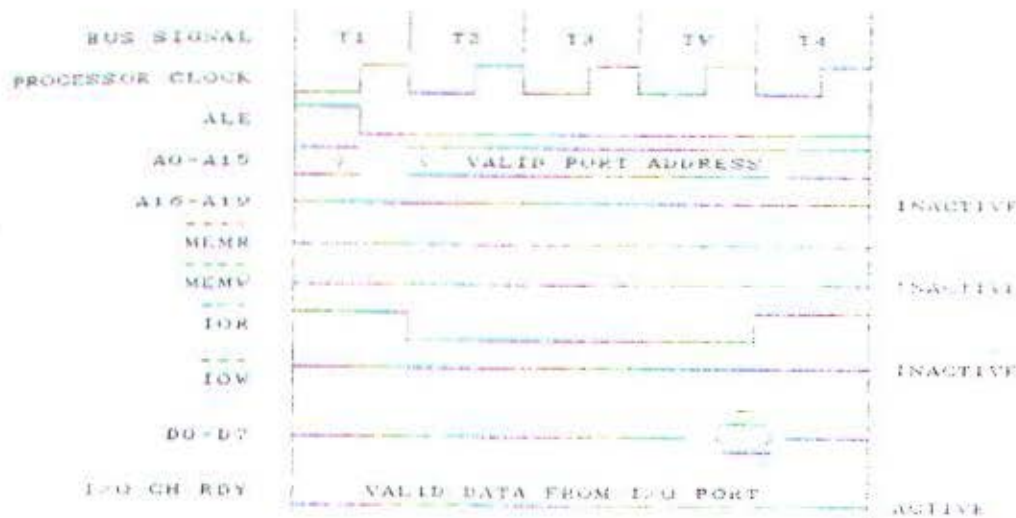
Gambar 2.2. Siklus Baca dari I/O Port ²⁾

2.2.3. Siklus Tulis dari I/O Port

Siklus ini aktif pada saat prosesor mengirimkan instruksi "OUT". Siklus ini digunakan untuk mengirimkan data ke alamat port dari data bus. Siklus waktu operasi tulis ini hampir sama seperti pada operasi baca dari I/O port. Perbedaannya adalah pada siklus T2 prosesor mengirimkan sinyal IOW dan data siap ditransfer ke halaman port. Pada saat siklus T4,

²⁾ Ibid, hal. 47.

IOW dinonaktifkan dan data yang benar masuk kealamat port. Pada gambar dibawah ini ditunjukkan sinyal T_w disisipkan secara otomatis oleh prosessor, sehingga keseluruhan operasi tulis ini memerlukan waktu ± 5 clock (75ns).



Gambar 2.3. Siklus Tulis dari I/O Port ³⁾

2.2.4. Pemetaan I/O Port IBM PC

Tabel dibawah ini menunjukkan pemetaan alamat I/O yang disediakan untuk sistim board dan slot I/O. Alamat 00H sampai FFH disediakan untuk sistim board dan tidak dapat dipakai untuk interfacing. Sedangkan alamat 100H sampai 3FFH disediakan pada slot I/O. Dari alamatport yang disediakan pada slot I/O terdapat ruang kosong yang dapat dipakai interfacing.

³⁾ Ibid., hal 47.

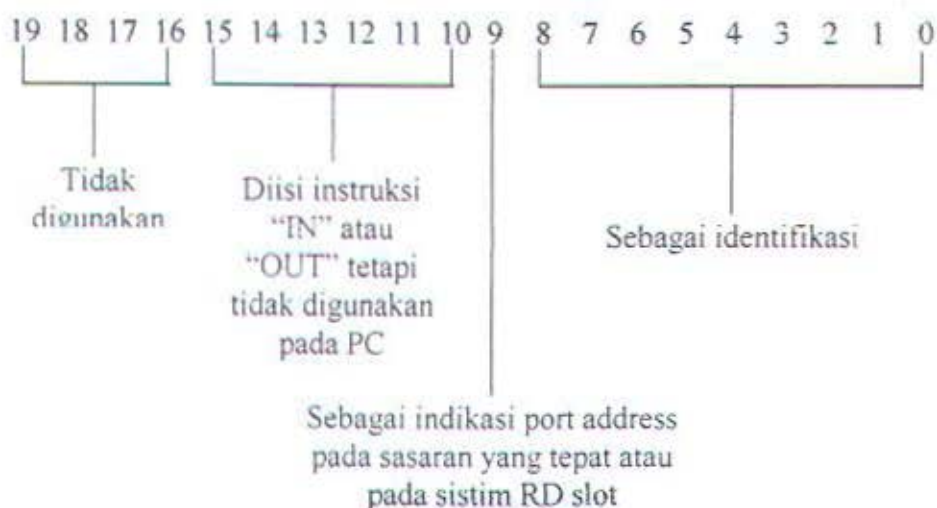
Tabel 2.1.

Pemetaan Alamat Port I/O IBM PC

| | |
|-------------|--|
| 000H - 01FH | DMA controller 1 |
| 020H - 03FH | Interrupt Controller 1 |
| 040H - 05FH | Timer |
| 060H - 06FH | Keyboard |
| 070H - 07FH | Real-time Clock NMI Mask |
| 080H - 09FH | DMA Page Register |
| 0A0H - 0BFH | Interrupt Controller 2 |
| 0C0H - 0DFH | DMA controller 2 |
| 0F0H | Clear Math Coprocessor Busy |
| 0F1H | Reset Math Coprocessor |
| 0F8H - 0FFH | Match Coprocessor |
| 0F0H - 1F8H | Fixed Disk |
| 200H - 207H | Game I/O |
| 278H - 27FH | Pararel Printer Port 2 |
| 278H - 2FFH | Serial Port 2 |
| 300H - 31FH | Prototype Card |
| 360H - 36FH | Cadangan |
| 378H - 37FH | Pararel Printer Port 1 |
| 380H - 38FH | SDLC, Bisynchronous 2 |
| 3A0H - 3AFH | Bisynchronous 1 |
| 3B0H - 3BFH | Monochrome Display dan Printer Adapter |
| 3COH - 3CFH | Cadangan |
| 3DOH - 3DFH | Color/Graphics Monitor Adapter |
| 3F0H - 3F7H | Diskette Controller |
| 3F8H - 3FFH | Serial Port 1 |

2.2.5. Address Decoder

Agar komputer dapat berkomunikasi dengan peralatan luar maka diperlukan alamat port, dengan melalui instruksi IN dan OUT data dapat dikirim ke atau dari port tersebut. Dari arsitektur PC dapat diakses alamat sebanyak 1.048.576 alamat, sebab mempunyai 20 line. Tetapi dalam perencanaan I/O dari PC tidak semua alamat yang digunakan. Dari ke-20 line ini A_9 mempunyai fungsi khusus, yaitu apabila bit ini pada logika "0" data tidak dapat diterima atau dikirim pada data bus dari card slot, transfer data hanya bisa dilakukan didalam sistem board. Oleh sebab itu untuk mengakses data dari port bit ke 9 (A_9) harus pada logika "1". Dari ke sepuluh bit alamat ini dapat diakses sebanyak 1024 alamat tetapi yang dapat digunakan untuk akses ke I/O port hanya 512 alamat. Seperti yang ditunjukkan gambar dibawah ini.



Gambar 2.4. Konfigurasi address untuk I/O port ⁴⁾

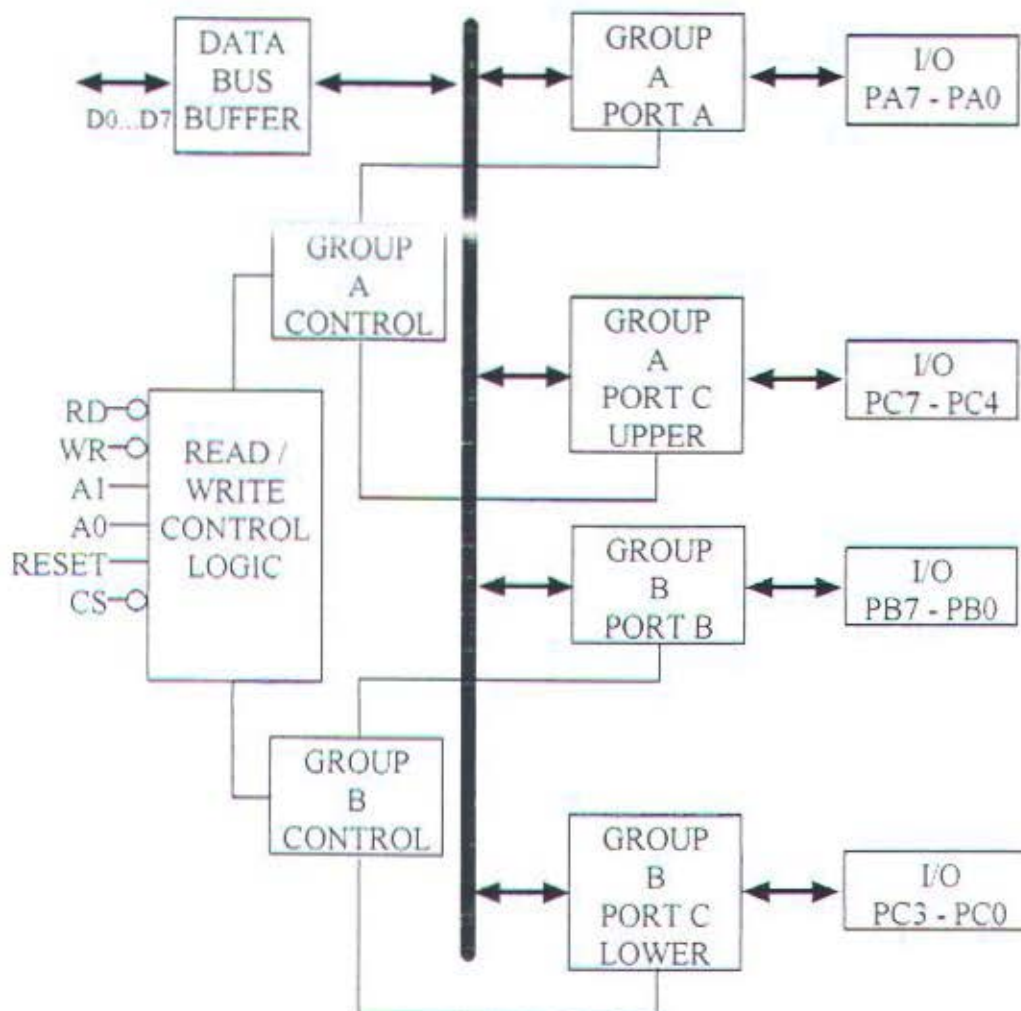
⁴⁾ Ibid, hal. 126.

2.3. Programmable Peripheral Interface 8255.

PPI atau Programmable Peripheral Interface merupakan perangkat antar muka yang dapat diprogram. PPI yang umum menggunakan chip 8255, melewati rangkaian PPI dengan menggunakan sebuah PC dapat dilakukan sejumlah pengaturan pekerjaan. Sebab rangkaian PPI dapat menghasilkan sejumlah perintah atas dasar sejumlah instruksi. Instruksi dapat dilakukan dari luar komputer atau dari dalam komputer atau juga dapat dari kedua-duanya (kombinasi perintah). Sehingga dapat dikatakan bahwa rangkaian PPI sanggup menghasilkan perintah-perintah universal dan dapat langsung diubah menjadi perintah-perintah konversi.

PPI 8255 memiliki 3 port yang dapat diprogram sebagai input/output port, yaitu port A, port B, port C. Ketiga port tersebut dibagi menjadi 2 group yaitu group A yang terdiri atas port A dan port C upper ($PC_3 - PC_0$), dimana masing-masing group diatur oleh rangkaian kontrol group yang bersangkutan.

Input jalur data pada PPI 8255 menggunakan buffer bidirectional internal yang bersifat tristate output dan masing-masing port yang bersifat latch, sehingga data yang dikeluarkan ke port akan tetap keadaannya selama tidak diubah atau chip direset. Gambar dibawah merupakan blok diagram dari PPI 8255, bagaimana kedudukan dari tiap-tiap port, kontrol dari masing-masing group serta buffer bidirectional untuk jalur data.



Gambar 2.5. Blok Diagram PPI 8255⁵⁾

2.3.1. Fungsi Pin-Pin Pada PPI 8255

Untuk mengetahui cara kerja dari PPI 8255, perlu diketahui blok fungsional yang terdapat didalam chip serta pin-pin yang terhubung pada setiap blok fungsional tadi. Beberapa pin yang akan dibahas adalah :

⁵⁾ Steeman J.P.M., Data Sheet Book 2, Elektuur-Elex, 1988, hal 240.

- RD : Logika '0' pada pin ini bersama-sama dengan logika '0' pada CS memungkinkan 8255 untuk mengirimkan data ke CPU. Jadi CPU melakukan instruksi pembacaan data dari PPI 8255.
- WR : Logika '0' pada pin ini bersama-sama dengan logika '0' pada CS memungkinkan CPU untuk mengirim data atau kontrol word ke PPI 8255.
- A₀-A₁ : Ada 4 elemen yang dapat dibentuk dari kombinasi A₀ dan A₁, jadi A₀ dan A₁ menentukan pemilihan salah-satu dari port dan satu kontrol register.
- RESET : Logika '0' pada input pin RESET mengakibatkan kontrol word register terhapus dan ketiga port diset sebagai mode input.
- PA₀-PA₇ : Jalur ini terdiri dari 8 jalur I/O port yang digunakan untuk penyambungan pada peralatan yang akan dikembangkan.
- PB₀ - PB₇ : Jalur sinyal ini juga merupakan jalur 8 bit I/O port yang digunakan untuk penyambungan peralatan yang akan dikembangkan,
- PC₀ - PC₇ : Jalur sinyal ini juga merupakan 8 bit I/O port yang digunakan untuk penyambungan peralatan yang akan dikembangkan. Selain itu juga jalur ini terbagi menjadi dua bagian dalam 4 bit yang digunakan untuk mengontrol jalur PB₀ - PB₇ dan jalur PA₀ - PA₇ dalam mode operasi 8255.



Gambar 2.6. PPI 8255 ⁶⁾

2.3.1. Mode Operasi pada PPI 8255

Pada PPI 8255, terdapat tiga macam mode operasi yang dapat diprogram ke masing-masing port yaitu :

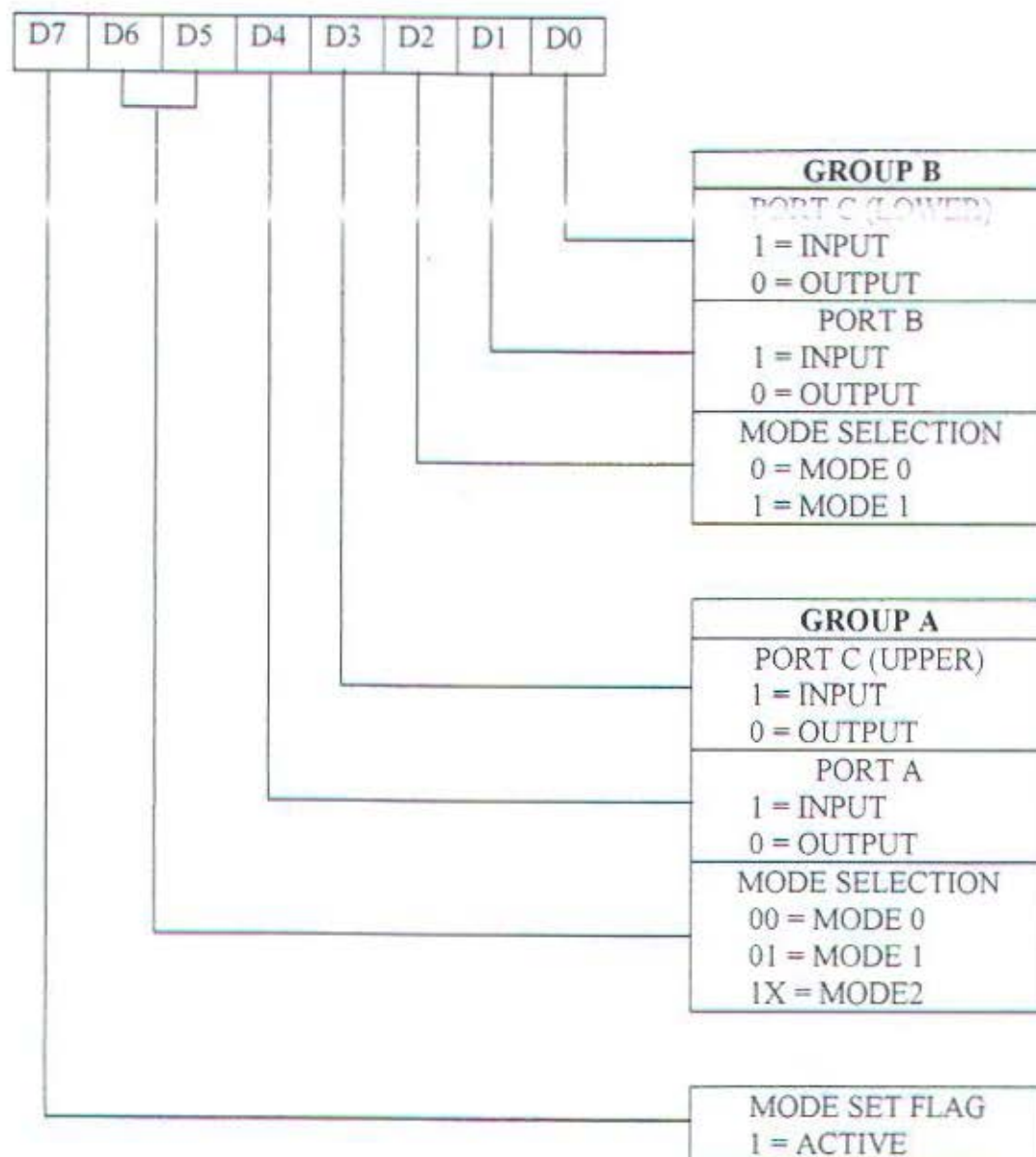
Mode 0 : Basic I/O, yaitu keadaan dimana port A, port B dan port C bekerja dengan sifat yang sama, yaitu Basic I/O, Output Latched dan Input Not Latched.

⁶⁾ Ibid., hal. 241.

Mode 1 : Strobed I/O, yaitu keadaan dimana port A dan port B digunakan sebagai I/O port, sedangkan port C digunakan sebagai Handshake lines.

Mode 2 : Bidirectional Bus, yaitu keadaan dimana port A digunakan sebagai port data dalam dua arah dengan 8 bit port kontrol ($PC_0 - PC_7$), sedangkan port B dapat dimanfaatkan pada mode 0 maupun mode 1.

Pemilihan mode tersebut dilakukan dengan cara mengisi suatu kontrol word di dalam kontrol word register. Format dari kontrol word terlihat seperti pada gambar dibawah. Dalam kontrol word tersebut berisi informasi tentang mode operasi yang diprogramkan untuk suatu port.



Gambar 2.7. Format Control Word Register PPI 8255⁷⁾

2.4. Konversi Data Analog Ke Digital

Untuk mengkonversikan input analog dari sensor ke sinyal digital agar dapat diproses oleh mikroprosesor IBM PC digunakan IC ADC.

⁷⁾ Ibid., hal. 242.

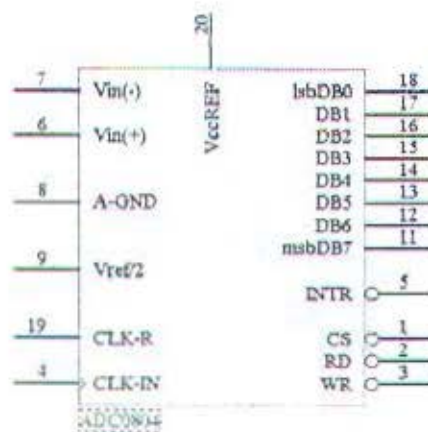
output 4 bit dan ADC dengan output 8 bit. Ada beberapa ADC yang mempunyai output 8 bit yang tersedia di pasaran antara lain ADC 0801, ADC 0802, ADC 0803, ADC 0804 dan ADC 0808. Untuk ADC 0801, ADC 0802, ADC 0803 dan ADC 0804 hanya memiliki sebuah input diferensial yaitu $V_{in}(+)$ dan $V_{in}(-)$. Lain halnya dengan keempat ADC tersebut ADC 0808 memiliki 8 channel input dengan 3 input select. Dalam hal banyaknya input ADC 0808 ini memenuhi kriteria untuk digunakan pada rangkaian akusisi data dalam tugas akhir ini, namun dalam hal tingkat resolusi outputnya ADC tersebut tidak sesuai. Oleh karena itu IC ADC yang digunakan dalam rangkaian akusisi data ini adalah ADC 0804 yang memiliki $V_{ref}/2$ sehingga lebih presisi.

2.4.1. Analog To Digital Converter 0804

Tipe Analog To Digital Converter (ADC) yang digunakan dalam tugas akhir ini adalah tipe ADC 0804. ADC yang digunakan ini merupakan ADC dengan data output 8 bit, dengan input tunggal. ADC yang digunakan ini mempunyai ketelitian sampai dengan 1 mV. Gambar dibawah akan memperlihatkan konfigurasi pin-pin dari ADC 0804. Secara umum IC ADC 0804 memiliki karakteristik sebagai berikut :

- Kompatibel dengan 0808 μP , dengan waktu akses 135ns.
- Dapat diinterfacekan dengan mikroprosesor, atau dapat dioperasikan secara terpisah.

- Logic input dan output dapat dioperasikan dengan level tegangan MOS dan TTL.
- Dapat beroperasi dengan (tegangan referensi / 2) sebesar 2.5V.
- Tegangan input analog memiliki jangkauan 0V - 5V dengan supply 5V.
- Tidak membutuhkan eksternal clock.
- Internal clock dengan R dan C.



Gambar 2.8. Susunan pin ADC 0804 ⁸⁾

2.5. Sensor Suhu

Berdasarkan jenis rangkaiannya sensor suhu dapat terbagi menjadi 4 tipe, yaitu Thermistor, Resistor peka suhu (PTC dan NTC), Integrated Circuit (IC) dan Thermocouple. Prinsip kerja keempat jenis sensor tersebut secara umum adalah mengubah besaran suhu menjadi

⁸⁾ National Semiconductor, Linear Data Book, Santa Clara, California, hal 3-24.

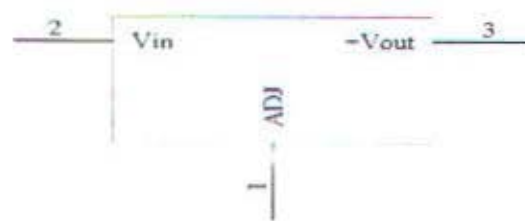
besaran tegangan. Masing-masing sensor suhu tersebut memiliki karakteristik yang berbeda-beda. Pada Thermistor panas yang diterima resistor berpengaruh terhadap harga resistansi sehingga menyebabkan perubahan tegangan pada resistor sehingga perubahan tegangan tersebut yang digunakan untuk pembacaan suhu. Pada sensor suhu resistor peka suhu ini terdiri dari dua macam yaitu PTC dan NTC.

Sedangkan sensor suhu yang digunakan dalam rangkaian akusisi data ini adalah sensor tipe yang ketiga dan yang terakhir yaitu Integrated Circuit yaitu IC LM35. Berbeda dengan ketiga jenis sensor suhu lainnya sensor IC ini memiliki keistimewaan yaitu memiliki karakteristik output yang linear. Pada sensor jenis IC ini ada 4 tipe yang tersedia di pasaran yaitu IC LM35, IC LM135, IC LM235 dan IC LM335. Masing-masing IC ini memiliki prinsip kerja yang sama hanya saja berbeda dalam hal jangkauan operasinya dan tingkat linearitas outputnya. Untuk IC LM35 memiliki karakteristik output yang lebih linear dibandingkan ketiga tipe IC lainnya dan kalibrasi output yang sesuai dalam skala derajat Celcius dengan koefisien perubahan tegangan terhadap suhu $10 \text{ mV}/^{\circ}\text{C}$. Berbeda dengan IC LM35, pada IC LM 135, 235, 335 kalibrasi output berdasarkan skala derajat Kelvin dengan koefisien perubahan tegangan terhadap suhu $10 \text{ mV}/^{\circ}\text{K}$. Ketiga IC tersebut beroperasi pada range $400\mu\text{A}$ - 5mA dengan impedansi output dinamic kurang dari 1Ω . Perbedaan dari ketiga IC tersebut adalah IC LM135 bekerja pada jangkauan ukur suhu -55°C

sampai dengan $+150^{\circ}\text{C}$, sedangkan IC LM235 bekerja pada suhu -40°C sampai dengan $+125^{\circ}\text{C}$ dan LM335 beroperasi pada suhu -40°C sampai dengan $+100^{\circ}\text{C}$.

2.5.1. IC LM35

Salah satu sensor suhu yang digunakan dalam sistem akuisisi data ini adalah IC LM 35. Sensor LM35 ini merupakan sensor suhu yang memiliki tingkat presisi yang tinggi dimana tegangan output yang dihasilkan sensor tersebut berbanding lurus dengan suhu dalam ukuran derajat Celcius (outputnya linear). Sensor suhu ini beroperasi pada range temperatur -55°C sampai $+150^{\circ}\text{C}$ dengan koefisien perubahan tegangan terhadap suhu sebesar $10\text{ mV}/^{\circ}\text{C}$ dengan akurasi $0,5^{\circ}\text{C}$. Sensor suhu ini memiliki keistimewaan yaitu impedansi output yang sangat kecil ($0,1\ \Omega$ untuk beban 1mA) dan kalibrasi hanya dilakukan sekali untuk selanjutnya perbandingan tegangan terhadap suhu tetap dan harganya lebih murah dibandingkan thermocouple. IC LM 35 ini mempunyai 3 kaki yaitu kaki tegangan supply ($V+$), kaki ground dan kaki output. Jangkauan ukur tegangan IC LM 35 ini 4 sampai dengan 30 V dengan arus kurang dari $60\mu\text{A}$.



Gambar 2.9 IC LM 35⁹⁾

2.6. Rangkaian Penguat Operasional

Untuk mempermudah analisa, biasanya op-amp diasumsikan sebagai op-amp ideal. Ada 3 daerah operasi op-amp, yaitu saturasi negatif, daerah linier dan saturasi positif.

Op-amp ideal memiliki karakteristik sebagai berikut :

- Penguatan ikal terbuka (A_{OL}) sama dengan tak terhingga.
- Tegangan keluaran $V_o = 0$ bila tegangan (+) input = tegangan (-) input.
- Impedansi masukan besarnya tak terhingga, sehinggataak ada arus pada terminal masukan.
- Impedansi keluaran = 0.
- Bandwidth tak terhingga.
- Common-mode Rejection Ratio (CMRR)tak terhingga, sehingga common-mode noise voltage = 0.
- Slew rate tak terhingga, sehingga perubahan tegangan output selalu sesuai dengan tegangan input.

⁹⁾ Opcit., Data Sheet Book 2, hal.142.

Karakteristik ideal ini penting untuk dipakai mengembangkan persamaan-persamaan rangkaian feedback dasar. Dengan bertitik tolak dari karakteristik op-amp ideal inilah diharapkan pemakai dapat menentukan dan memilih jenis op-amp yang dipakai dalam rangkaian-rangkaian yang dibuat.

Pada daerah saturasi, tegangan keluaran bukan merupakan fungsi linier dari tegangan masukan dan besarnya sekitar 90% dari tegangan catu yang digunakan. Artinya pada daerah saturasi negatif akan didapat tegangan saturasi negatif dan pada daerah saturasi positif akan didapat tegangan saturasi positif.

2.6.1. Penguat Operasional Sebagai Voltage Follower

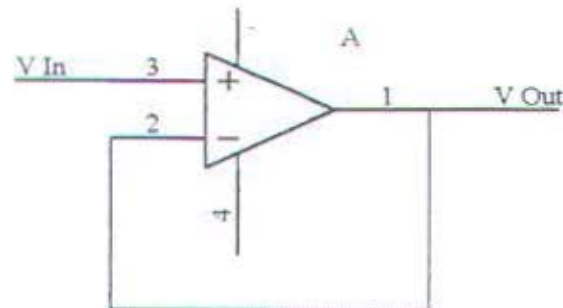
Voltage follower sering juga disebut source follower, unity gain amplifier, buffer amplifier, atau isolation amplifier.

Tegangan input E_i dipasang langsung pada input (+) karena tegangan antara pin (+) dan pin (-) dari op-amp itu dapat dianggap nol maka berarti $V_o = V_i$.

Tegangan output = tegangan input, baik besarnya maupun tandanya karena itu seperti namanya tegangan output mengikuti (follow) tegangan input.

$$V_o = E_i \dots \dots \dots (1)^{10)}$$

$$A_{cl} = V_o / E_i = 1 \dots \dots \dots (2)^{11)}$$



Gambar 2.10. Voltage Follower¹²⁾

2.6.2. Penguat Operasional Sebagai Penguat Pembalik (Inverting)

Rangkaian penguat operasional dapat digunakan sebagai penguat pembalik artinya penguatan yang dihasilkan berharga negatif. Hal ini berarti polaritas sinyal input berlawanan dengan sinyal output. Dalam rangkaian penguat pembalik ini gain penguatan diperoleh dari :

$$I = I_f \dots \dots \dots (3)$$

$$V_{in} / R_i = (0 - V_o) / R_f \dots \dots \dots (4)$$

$$V_o / V_{in} = - R_f / R_i \dots \dots \dots (5)$$

$$A = - R_f / R_i \dots \dots \dots (6)^{13)}$$

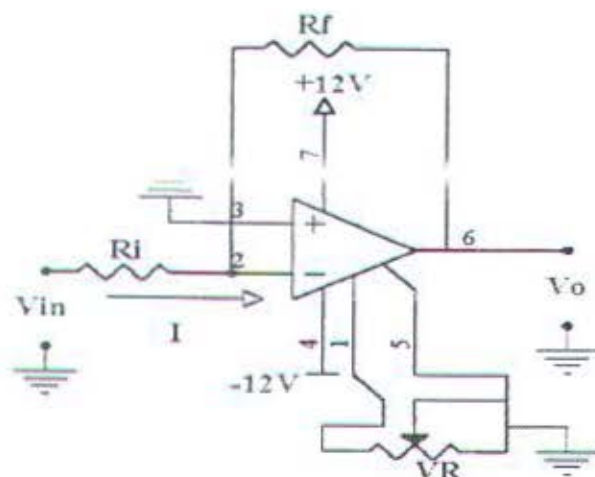
Dimana A = gain penguatan

¹⁰⁾ Coughlin, Robert F., Frederick F., Penguat Operasional dan Rangkaian Terpadu Linier, Herman Alih Bahasa, Erlangga, Jakarta, 1995, hal. 42.

¹¹⁾ Ibid., hal. 42.

¹²⁾ Ibid., hal. 42.

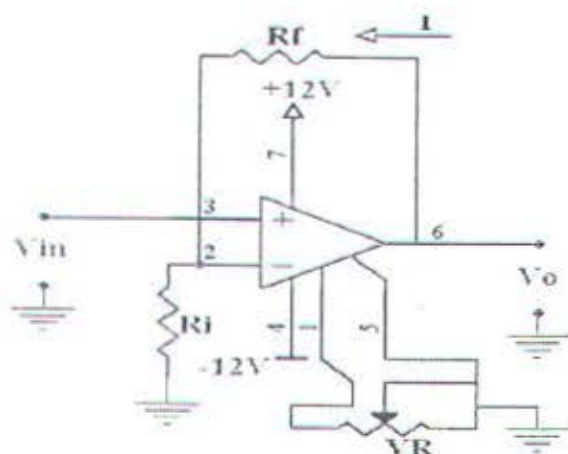
¹³⁾ Ibid., hal. 33.



Gambar 2.11. Rangkaian Penguat Pembalik¹⁴⁾

2.6.3. Penguat Operasional Sebagai Penguat Tak Membalik

Rangkaian penguat operasional dapat digunakan sebagai penguat tak-membalik, maksudnya sinyal keluaran yang dihasilkan polaritasnya sama dengan sinyal masukannya.



Gambar 2.12. Rangkaian Penguat Tak Membalik¹⁵⁾

¹⁴⁾ Ibid., hal. 33.

¹⁵⁾ Ibid., hal. 48.

Gain penguatan yang dihasilkan adalah sebagai berikut:

$$I = I_f \dots\dots\dots(7)$$

$$(0 - V_{in}) / R_i = (V_{in} - V_o) / R_f \dots\dots\dots(8)$$

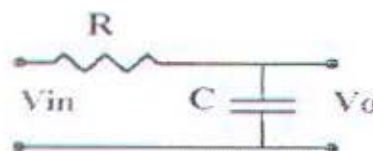
$$A = 1 + (R_f / R_i) \dots\dots\dots(9)^{16)}$$

2.7. Rangkaian Filter Pasif

Pada setiap pengukuran sinyal analog seringkali diperoleh gangguan atau noise yang menyebabkan sinyal output yang dihasilkan dalam pengukuran tidak sesuai dengan sinyal input sebenarnya. Untuk mengatasi masalah tersebut digunakan rangkaian filter. Berdasarkan prinsip kerjanya filter dibagi menjadi 4 macam, yaitu :

1. Low Pass Filter (LPF)
2. Band pass Filter
3. Band Elimination Filter
4. High Pass Filter (HPF)

Dalam proses akusisi data dengan mikroprosesor, filter ini mempunyai fungsi sebagai blokir frekuensi dan anti aliasing dimana pada rangkaian akusisi data dalam tugas akhir ini digunakan filter jenis Low Pass Filter.



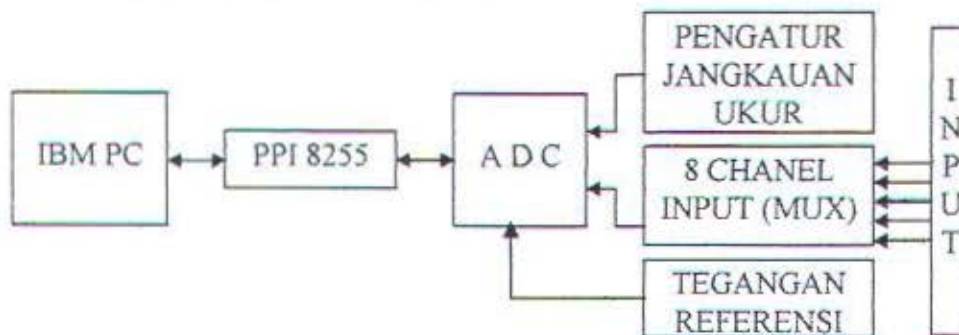
Gambar 2.13. Rangkaian Low Pass Filter

¹⁶⁾ Ibid., hal. 46.

BAB III

PERENCANAAN DAN PEMBUATAN PERANGKAT KERAS

3.1. Blok Diagram Rangkaian Akusisi Data



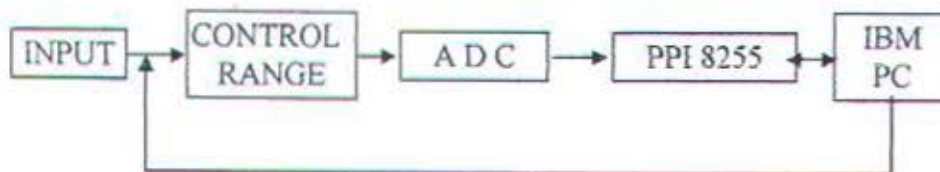
Gambar 3.1.

Blok Diagram Keseluruhan Rangkaian Akusisi Data

Blok diagram rangkaian akusisi data yang direncanakan adalah seperti ditunjukkan pada gambar 3.1. Rangkaian ini terdiri dari seperangkat IBM PC yang didalamnya terhubung interface card. Interface tersebut dihubungkan ke rangkaian ADC yang didalamnya terdapat rangkaian pengatur jangkauan ukur, rangkaian channel input dan rangkaian pembangkit tegangan referensi. Dari rangkaian ADC dihubungkan ke input analog melalui selektor dimana rangkaian ini mempunyai 8 saluran input. Rangkaian pengatur jangkauan ukur dihubungkan pada kaki tegangan input ADC (V^+), sedangkan rangkaian pengatur channel yang berfungsi untuk memilih input yang diinginkan dihubungkan pada kaki (V_{in-}) ADC. Dalam sistem ini rangkaian interface yang digunakan adalah PPI 8255.

3.2. Prinsip Kerja Rangkaian Akusisi Data

Prinsip kerja rangkaian akusisi data secara keseluruhan dapat terlihat pada gambar dibawah ini.



Gambar 3.2.

Blok Diagram Sistematika Kerja Rangkaian Akusisi Data

Dari blok diagram tersebut dapat dijelaskan sebagai berikut :

1. Dari modul input yang ada dihasilkan tegangan yang berbentuk analog dimana output tersebut bergantung pada modul input yang ada, misalnya untuk sensor suhu LM35 akan menghasilkan tegangan sesuai dengan suhu yang diterima sensor tersebut. Kemudian tegangan tersebut dikirimkan ke channel input dimana jangkauan tegangan tersebut antara 0 - 2V.
2. Pada channel input terpilih data yang terbaca dikirimkan ke ADC setelah sebelumnya dilewatkan buffer. Range tegangan pada control range dipasang pada posisi awal (terkecil).
3. Di ADC data tegangan berbentuk sinyal analog tersebut dikonversi ke data digital agar dapat dibaca IBM-PC, kemudian dikirimkan ke PPI Card.

4. Melalui PPI Card data berbentuk digital tersebut dibaca oleh IBM-PC untuk kemudian dibandingkan apakah data tersebut sama besar atau melebihi range yang ditetapkan. Bila melebihi atau sama dengan range yang ditetapkan maka komputer akan merubah control range satu tingkat diatas range awal melalui PPI. Proses ini berlangsung hingga range tegangan sesuai.
5. Kemudian data yang sudah benar tersebut ditampilkan dalam layar monitor. Dalam sistim ini sampling data dilakukan dengan menggunakan komputer/software komputer.
6. Proses ini berlanjut untuk channel input yang lainnya.

3.3. Rangkaian Interface PPI 8255

Dalam rangkaian PPI 8255 ini menggunakan address decoder untuk penempatan I/O masing-masing port seperti terlihat pada tabel dibawah ini.

Tabel 3.1.

Tabel Address Decoder Untuk Alamat 300H - 303H.

| Port | Address | A ₉ | A ₈ | A ₇ | A ₆ | A ₅ | A ₄ | A ₃ | A ₂ | A ₁ | A ₀ |
|---------|---------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| Port A | 300H | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Port B | 301H | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Port C | 302H | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Port CW | 304H | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

Hasil Output dari address decoder tersebut merupakan sinyal pilih yang dihubungkan ke chip select dari rangkaian interface (PPI 8255), sehingga rangkaian interface terbut dapat bekerja jika address yang dipilih sesuai dengan address decodernya. Dari tabel address decoder diatas dapat dibuat rangkaian untuk address decoder dengan perencanaan sebagai berikut :

- Output A_1 dan A_0 dari IBM PC dihubungkan pada kaki input PPI 8255 untuk menentukan port yang akan digunakan.
- A_2 , A_3 dan A_8 dihubungkan pada input decoder 74LS138 dan output yang dihasilkan untuk menentukan alamat yang akan dipakai dengan ketentuan sebagai berikut :

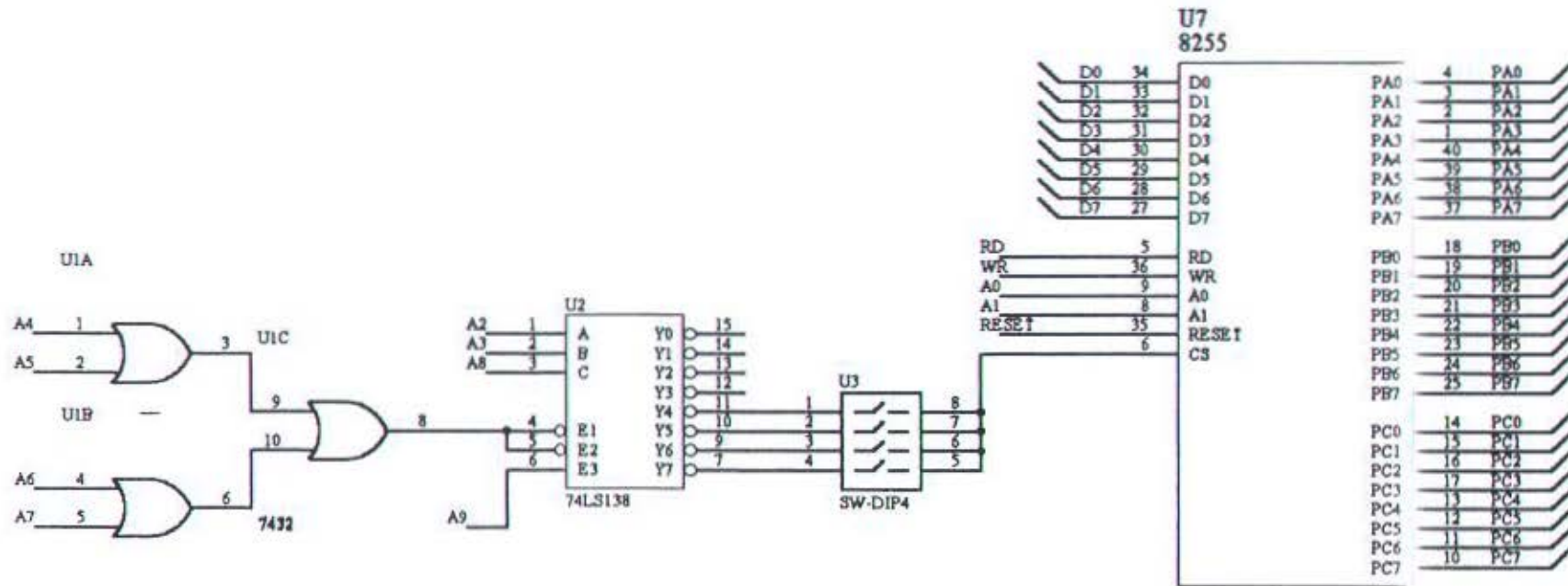
Tabel 3.2.

Penentuan Alamat Pada Input - Output Decoder

| A_8 | A_3 | A_2 | Output | Alamat |
|-------|-------|-------|--------|-------------|
| 1 | 0 | 0 | Y4 | 300H - 303H |

- $A_4 \dots A_7$ dan A_9 digunakan untuk input enable pada decoder 74LS138, $A_4 \dots A_7$ untuk input enable dengan input low dan A_9 untuk input enable dengan input high dengan rangkaian keseluruhan address decoder sebagai berikut :

Output dari rangkaian interface PPI 8255 ini terdiri dari 3 port yaitu port A, port B dan port C dimana port C dibagi menjadi 2

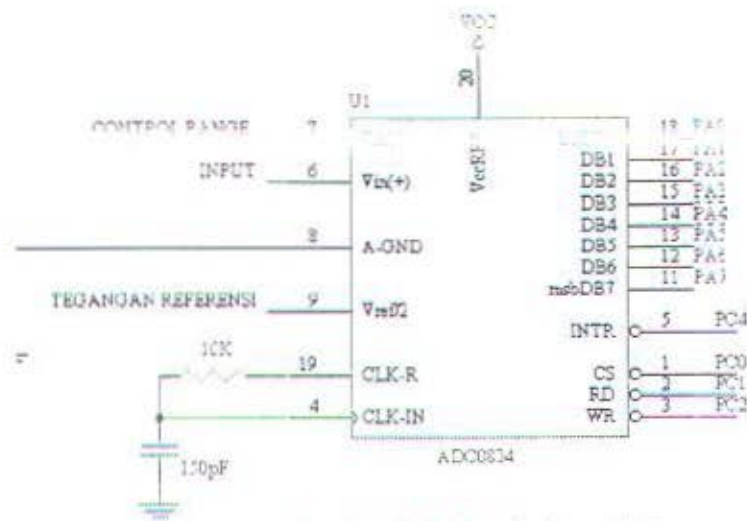


Gambar 3.3. Rangkaian PPI 8255

yaitu port C upper dan port C lower. Pada rangkaian ini port yang digunakan adalah port A untuk input, port B untuk output dan port C upper sebagai input dan port C lower sebagai output.

3.4. Rangkaian Konversi Data Analog ke Digital

Mikroprosesor IBM PC hanya dapat memproses sinyal-sinyal digital oleh karena itu agar tegangan searah yang merupakan keluaran dari rangkaian-rangkaian sensor dapat diolah oleh mikroprosesor IBM PC harus diubah terlebih dahulu ke bentuk digital dengan rangkaian ADC. Dalam perencanaan digunakan ADC 0804 yang memiliki 8 bit data sesuai dengan mikroprosesor IBM PC. Pada rangkaian ADC ini kaki VI(-) dihubungkan dengan rangkaian pengontrol jangkauan ukur. Kaki VI(+) dihubungkan dengan rangkaian pengatur channel input yang telah dibuffer terlebih dahulu. Rangkaian ADC ini memperoleh tegangan referensi sebesar 128mV yang diperoleh dari rangkaian pembangkit tegangan referensi. Frekuensi clock ADC 0804 sebesar 1 Mhz yang dihubungkan pada pin CLKR dan CLK. Pin-pin control ADC 0804 yaitu CS, RD, dan WR dihubungkan ke kontrol bus sistem mikroprosesor IBM PC yang menuju ADC.



Gambar 3.4. Rangkaian ADC

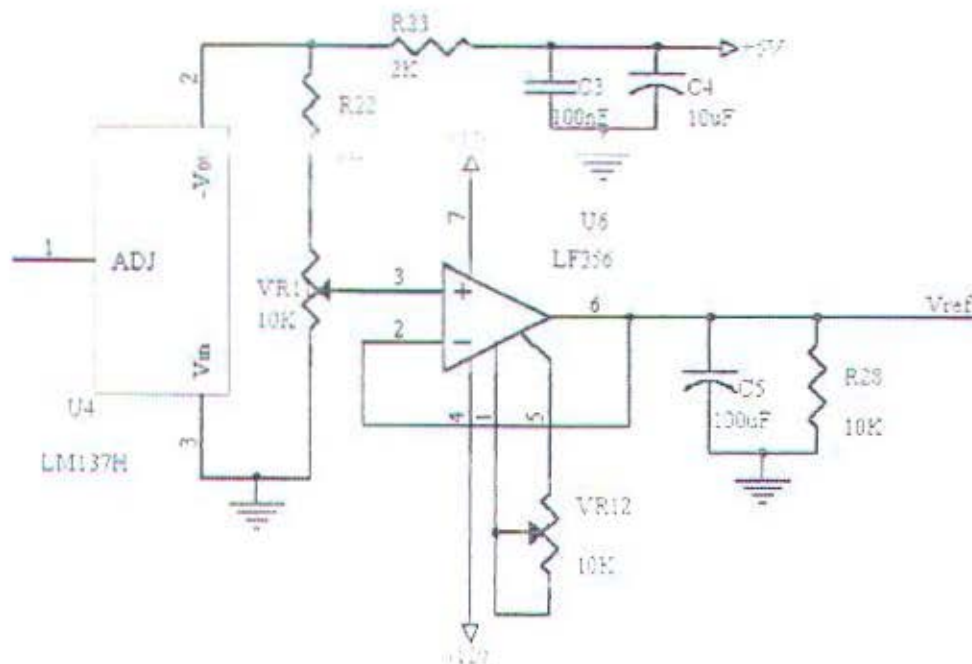
3.5. Rangkaian Pembangkit Tegangan Referensi

Pembangkit tegangan referensi berfungsi untuk menghasilkan tegangan ADC yang dibutuhkan untuk dapat melakukan konversi. Tegangan referensi yang dipakai untuk konversi adalah 256mV. Pin Vref dari ADC hanya memerlukan tegangan setengah dari tegangan referensi, sehingga pembangkit tegangan harus membangkitkan tegangan sebesar 128 mV. Tegangan referensi ini harus diusahakan benar-benar DC murni, karena jika tidak murni atau ada unsur AC/ripple, akan menyebabkan error dalam konversi sinyal dari analog ke digital.

Rangkaian lengkap dari pembangkit gelombang referensi diperlihatkan pada gambar dibawah. Untuk menghasilkan tegangan terregulasi dipakai IC LM336. IC ini merupakan reference voltage dengan

output yang dihasilkan adalah stabil 2,5V. Pada IC ini terdapat pin adjust, untuk mengatur output tegangan yang dihasilkan pada sekitar 2,5V.

Karena tegangan referensi yang diinginkan adalah 128mV, maka output dari IC LM336 diturunkan dengan menggunakan suatu voltage divider yang dibentuk dari sebuah resistor dan potensiometer. Untuk mendapatkan ketelitian yang baik, untuk potensiometernya dipilih dari jenis multiturn 10K. Potensiometer diset sampai dihasilkan tegangan referensi yang dibutuhkan yaitu 128 mV. Tegangan hasil pembagian ini kemudian dibuffer dengan suatu voltage follower yang dibentuk dari IC LF356. IC ini mempunyai konfigurasi FET pada tahap inputnya, sehingga impedansi inputnya sangat tinggi. Dengan impedansi yang tinggi ini, tidak akan terjadi efek pembebanan pada input. Untuk menekan sekecil mungkin ripple yang mungkin ada output dari voltage follower diberi filter RC paralel. Setelah difilter baru dihubungkan dengan input reference voltage, yaitu pin no. 9 dari ADC 0804.

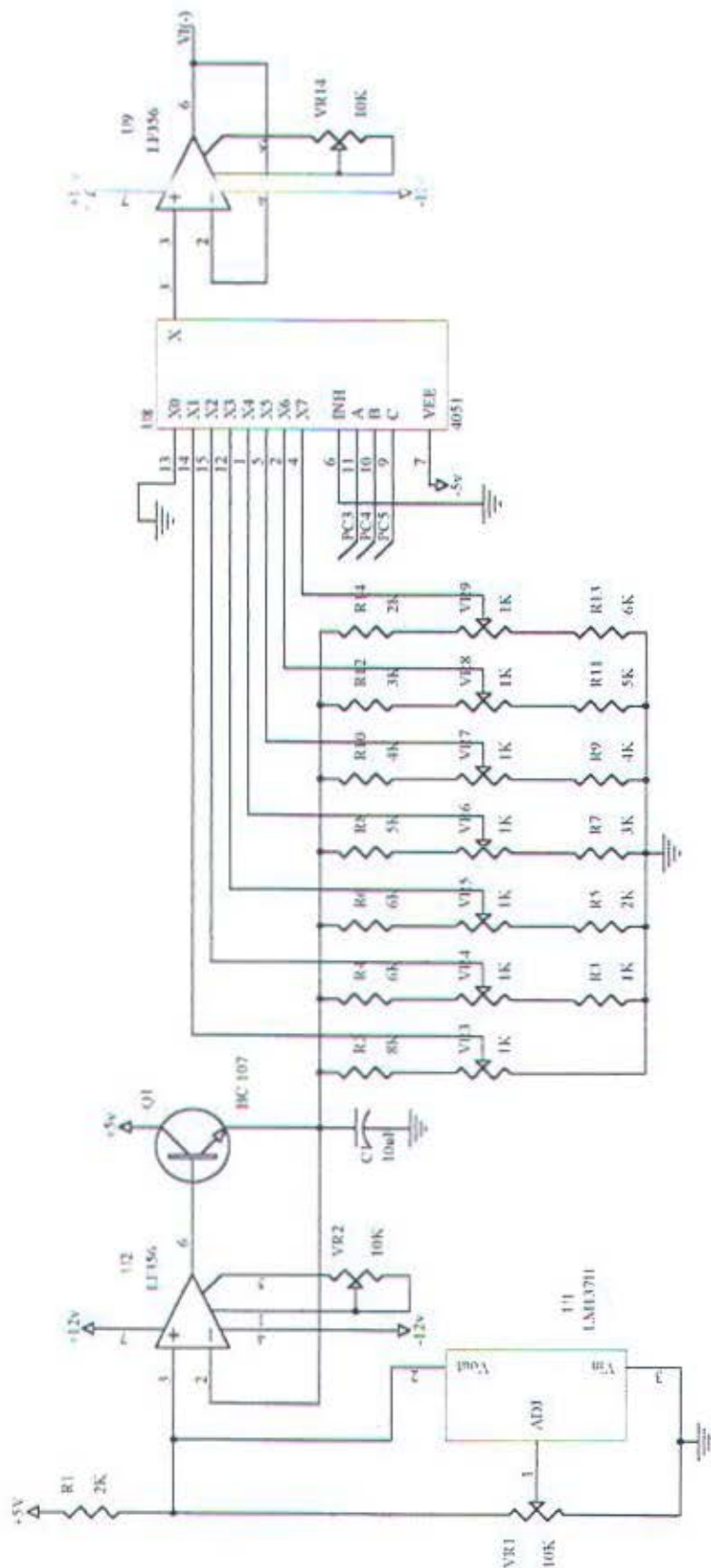


Gambar 3.5. Rangkaian Tegangan Referensi

3.6. Rangkaian Pengatur Jangkauan Ukur

Dalam keadaan biasa dengan sinyal referensi 128 mV, ADC 0804 hanya mampu mengkonversi tegangan maksimum sampai dengan 256 mV. Diatas tegangan 256 mV, ADC tidak akan mampu membedakannya lagi (overflow).

Dengan suatu teknik khusus, jangkauan dari ADC dapat diperluas sampai 2,0 V, tanpa mengurangi ketelitian dari hasil konversi, ketelitian tetap 1mV. Perluasan jangkauan tegangan diperoleh dengan suatu rangkaian yang disebut dengan pengontrol jangkauan. Gambar rangkaian pengontrol jangkauan diperlihatkan dalam gambar dibawah ini.



Gambar 3.6. Rangkaian Pengatur Jangkauan Ukur

Prinsip dasar dari pengontrol jangkauan adalah dengan mengubah-ubah besarnya tegangan yang masuk input VI-. Sinyal input yang akan dikonversi dimasukkan ke dalam VI+. Rangkaian pengontrol jangkauan akan menaikkan besarnya tegangan pada input VI- seandainya hasil yang dikonversi overflow. Rangkaian pengontrol tegangan mampu mengeluarkan tegangan teregulasi sebanyak 8 macam, yaitu 0V, 256 mV, 512 mV, 768 mV, 1024 mV, 1280 mV, 1536 mV, 1792 mV. Pemilihan tegangan referensi dilakukan dengan sebuah analog multiplexer 4051, yang mempunyai 8 input. Proses pemilihan input analog pada multiplexer dikontrol oleh komputer. Jadi mula-mula membaca data hasil konversi dari ADC, jika data itu ternyata overflow, maka akan dikirim sinyal untuk menaikkan tegangan pada input VI-. Proses dilakukan terus-menerus sampai hasil konversi tidak overflow. Jika hasil konversi tetap overflow, dan tegangan pada VI- sudah maksimum, berarti sinyal yang diukur sudah melewati jangkauan maksimum yang diperkenankan. Jangkauan maksimum adalah 2,0V.

Untuk menghasilkan tegangan yang teregulasi, dipakai IC LM336 seperti yang dipakai pada pembangkit tegangan referensi. Tegangan output IC ini stabil pada 2,5V. Karena banyak pembagi tegangan, maka output LM336 dibuffer terlebih dahulu dengan menggunakan IC LF356 dan transistor BC107. Dari sini kemudian dihubungkan dengan 7 buah pembagi

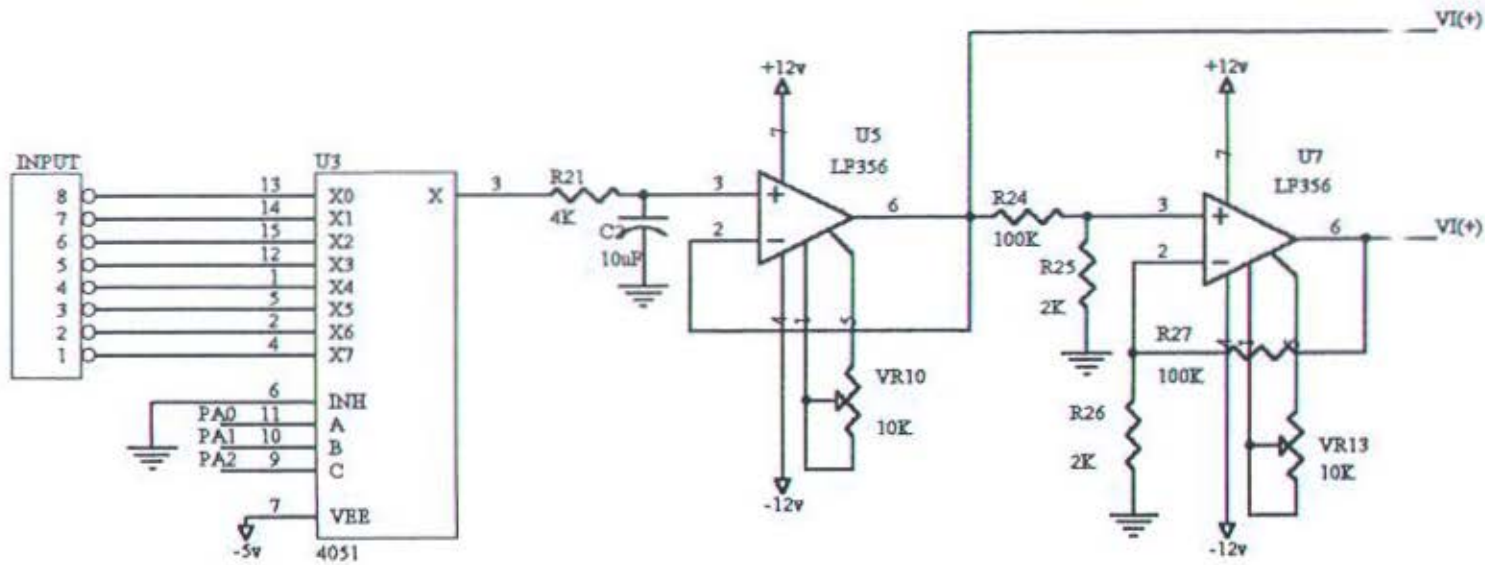
tegangan yang dibentuk dari kombinasi resistor dan potensiometer untuk menghasilkan tegangan input V_{I-} yang presisi.

Delapan macam tegangan tadi dimasukkan ke analog multiplexer 4051. Setelah dibuffer terlebih dahulu outputnya dengan IC LF 356 kemudian dihubungkan dengan V_{I-} atau pin no. 7 dari IC ADC 0804.

3.7. Rangkaian Pengatur Channel Input

Rangkaian pengontrol sinyal input berfungsi untuk memilih input mana dari 8 input yang ada, yang akan dikonversikan dari analog menjadi digital. rangkaian pengontrol ini memilih input yang aktif dengan menggunakan IC analog multiplexer 4051, sama seperti yang digunakan pada pengontrol jangkauan. Proses pemilihan dengan komputer melalui software. Gambar rangkaian pemilihan sinyal input diberikan dalam gambar 3.7.

Output dari analog multiplexer dibuffer dahulu sebelum dihubungkan ke input V_{+} dari ADC 0804. Buffer yang dipakai adalah IC LF356 dalam konfigurasi sebagai voltage follower. Pada rangkaian juga terdapat sebuah switch untuk memilih apakah sinyal harus melewati buffer kedua, ataukah langsung ke input V_{+} . Dari hasil pengukuran akan dilihat perbedaan dari kedua kondisi ini.



Gambar 3.7. Rangkaian Pengatur Chanel Input

3.8. Rangkaian Pengukur Suhu

Rangkaian input analog yang digunakan dalam sistem akuisisi data ini adalah sebuah rangkaian sensor suhu dan sebuah rangkaian pembagi tegangan potensiometer. Pada rangkaian sensor suhu yang digunakan dalam akuisisi data ini adalah rangkaian sensor suhu dengan menggunakan IC LM 35.

3.8.1. Rangkaian Sensor Suhu IC LM 35

Rangkaian Sensor suhu ini terdiri dari sebuah IC LM 35, sebuah resistor metal film bernilai $56K\Omega$ dan sebuah variabel resistor $200K\Omega$. Rangkaian ini dapat mengukur suhu ruangan dengan jangkauan temperatur $-55^{\circ}C$ sampai dengan $+150^{\circ}C$. Pemilihan harga resistor yang digunakan didasarkan pada :

$$R = -V_s / 50\mu A$$

dimana R = resistor dan V_s = sumber tegangan yang digunakan, sedangkan $50\mu A$ adalah syarat arus yang harus dipenuhi untuk IC LM35. Dalam rangkaian sensor suhu ini sumber tegangan yang digunakan adalah $+12V$ langsung ke sensor dan $-12V$ terhubung ke output melalui resistor. Dari ketentuan diatas diperoleh :

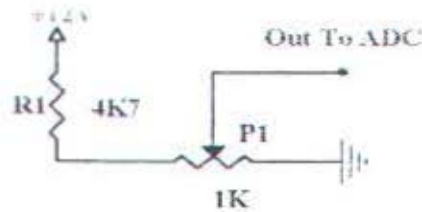
$$\begin{aligned} R &= 12V / 50\mu A \\ &= 240K\Omega \end{aligned}$$

Dalam rangkaian R dibentuk oleh resistor $56K$ dan variabel resistor $200K\Omega$.

3.9. Rangkaian Pembagi Tegangan Sebagai Input Chanel

Salah satu sinyal input yang digunakan dalam rangkaian akuisisi data ini dihasilkan dari rangkaian pembagi tegangan dengan potensiometer. Pada rangkaian pembagi tegangan ini tegangan yang dihasilkan hanya sampai 2 Volt. Rangkaian ini terdiri dari sebuah resistor dan sebuah potensiometer (singleturn). Sumber tegangan yang digunakan dalam rangkaian tersebut +12V. Fungsi dari resistor R1 adalah untuk mengurangi tegangan dari sumber agar tegangan yang dihasilkan yang dihasilkan potensiometer hanya mencapai 2 Volt atau sesuai dengan range tegangan input yang dapat dibaca ADC. Sinyal tegangan input yang

diukur adalah sesuai dengan besar harga resistansi potensiometer yang diputar secara manual.



Gambar 3.9, Rangkaian Pembagi Tegangan

Pemilihan harga resistor dan potensiometer yang digunakan adalah sesuai dengan persamaan berikut ini :

$$V_s = V_{P1} + V_{R1} \dots \dots \dots (10)$$

$$V_s = (i \cdot P1) + (i \cdot R1) \dots \dots \dots (11)$$

dimana $i = 2 \text{ mA}$ dan $V_s = V_{\text{supply}} = +12\text{V}$.

Agar harga tegangan maksimum yang dihasilkan P1 mencapai 2 Volt maka V_{R1} harus berharga atau mendekati harga 10 Volt yaitu dengan memberikan resistor metal film $4\text{K}7\Omega$. Selanjutnya harga potensiometer ditentukan dengan persamaan :

$$P1 = V_{P1} / i \dots \dots \dots (12)$$

$$P1 = 2 \text{ Volt} / 2\text{mA} = 1\text{K}\Omega$$

BAB IV

PERENCANAAN PERANGKAT LUNAK

Dalam menjalankan sistem akuisisi data ini tidak hanya dibutuhkan perangkat keras namun juga diperlukan perangkat lunak yang berfungsi mengatur jalannya sistem sekaligus menampilkan data yang diambil dari input-input atau sensor. Bahasa pemrograman yang digunakan dalam menjalankan sistem tersebut adalah bahasa Turbo Pascal versi 7.0. Di dalam pemrograman ini akan mengolah dan menampilkan data dari channel input ke layar monitor dalam bentuk tabel data numerik. Adapun algoritma dari pemrograman tersebut adalah sebagai berikut :

1. Inisialisasi Port Programmable Peripheral Interface (PPI 8255).

Dalam proses ini control word PPI 8255 diatur agar port A dan port C Upper dapat digunakan sebagai input, sedangkan port B dan port C Lower berfungsi sebagai output. Untuk itu control word PPI diset menjadi 98 atau 98H. Mode PPI 8255 yang digunakan dalam pemrograman ini adalah mode 0 dimana data dikirim dan dibaca langsung oleh tiap-tiap port tanpa memerlukan sinyal Strobe. Pada proses inisialisasi ini Port C diatur pada kondisi awal yaitu 0FH atau 0FH agar kaki CS, RD, dan WR pada ADC ada pada kondisi high sebagai tanda konversi data belum dimulai.

2. Pemilihan Chanel Input dan Menu Utama.

Pada menu utama ini terdapat proses penawaran pada pengguna untuk memilih chanel input yang diinginkan / dibaca datanya. Menu ini berupa suatu tampilan awal program dengan frame yang digabung dengan pemilihan chanel input dimana pemilihannya dengan menekan tombol chanel yang diinginkan.

3. Pengesetan Range Awal Tegangan Input.

Pada kondisi awal, setiap pembacaan data harus menggunakan range tegangan input 255mV. Tujuannya agar mudah untuk mendeteksi apakah tegangan sinyal input dari sensor melebihi range atau tidak.

4. Pengiriman Sinyal Start Konversi.

Setelah chanel input dipilih, agar dapat dilakukan pengkonversian data analog ke bentuk data digital maka diaktifkan sinyal enable yaitu WR, RD dan CS pada ADC dan menunggu pengiriman sinyal INT dari ADC. Pengaktifan sinyal-sinyal tersebut berdasarkan pemrograman interfacing yaitu dengan memberikan harga 00H pada Port C

5. Pembacaan Data ADC.

Pembacaan data dilakukan dengan membaca masukan dari Port A yang telah diatur sebagai saluran input melalui pemrograman.

6. Proses Pengkondisian Data dengan Maksimum Range Tegangan Input.

Data yang dibaca dan diolah komputer melalui port PPI tidak langsung ditampilkan hasilnya dalam layar monitor namun terlebih dahulu

melalui pengkondisian 'IF THEN ELSE' untuk dibandingkan dengan tegangan maksimum input dalam range awal. Bila data yang diterima komputer sama dengan nilai tegangan maksimum range awal maka akan nilai range tegangan input dinaikkan 1 tingkat dari range awal sedangkan bila data tidak sama dengan nilai maksimum range awal maka hasil data akan ditampilkan pada layar monitor.

7. Proses Pengubahan Range Tegangan Input.

Pada proses ini range tegangan input dinaikkan 1 tingkat dengan mengatur agar Port B5 dan Port B4 tetap berharga '0' sedangkan Port B3 diaktifkan atau diberikan harga '1'. Proses ini dapat berlanjut terus bila data masih melebihi nilai maksimum range berikutnya hingga data tersebut tidak sama dengan nilai maksimum range dan tegangan input yang dapat dibaca dibatasi hingga 2 Volt.

8. Menampilkan Hasil Pembacaan Data.

Data yang telah diolah dengan benar oleh komputer, hasilnya ditampilkan pada layar monitor dalam bentuk tabel numerik. Data tersebut ditampilkan dalam bentuk konversi tegangan dengan parameter data yang diukur misalnya suhu. Data yang ditampilkan dalam setiap tabel tersebut adalah sebanyak sepuluh data.

9. Pengkondisian Pengulangan Pembacaan Data .

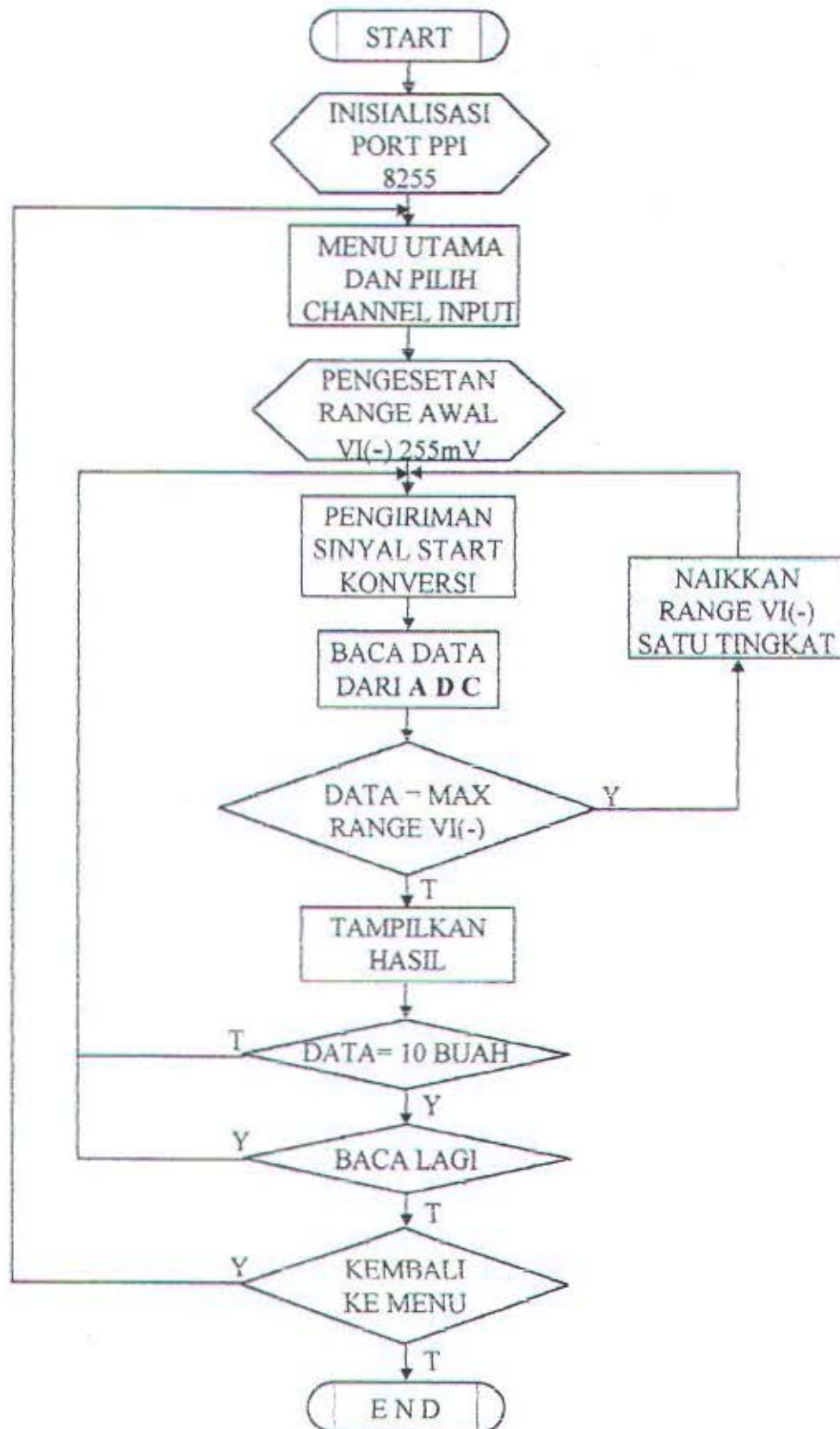
Pada pengkondisian ini merupakan penawaran apakah ingin membaca / menyampling data lagi dimana jika 'Ya' maka akan dilakukan

proses pembacaan data lagi pada chanel input yang sama. Bila 'Tidak' maka komputer akan melanjutkan proses berikutnya yaitu Pengkondisian untuk kembali ke tampilan menu.

10. Pengkondisian Kembali Ke Menu Utama.

Pada bagian ini berupa penawaran untuk kembali ke menu utama dan bila tidak ingin kembali ke menu utama maka proses selanjutnya adalah komputer akan mengakhiri program.

Bentuk algoritma pemrograman dalam tugas akhir ini dapat digambarkan dalam flowchart di bawah ini.



Gambar 4.1. Flowchart

BAB V

PENGUKURAN DAN PENGUJIAN ALAT

5.1. Pengukuran Rangkaian Input ADC

Setelah dilakukan perencanaan dan pembuatan alat maka langkah berikutnya adalah melakukan pengukuran dan pengujian peralatan. Pada pengukuran rangkaian sensor suhu ini dilakukan dengan memasukkan rangkaian sensor ke dalam ruangan sehingga yang diukur adalah suhu kamar yang berkisar antara 25°C - 32°C . Dalam pengukuran ini dibutuhkan sebuah voltmeter dan sebuah termometer untuk mengukur suhu dan tegangan. Tegangan yang diukur dikalibrasikan dengan suhu sebenarnya sehingga kenaikan tegangan sebanding dengan kenaikan suhunya. Untuk pengukuran rangkaian input lainnya yaitu pembagi tegangan dilakukan pengukuran harga resistansi potensiometer (single turn) kemudian dibandingkan dengan harga tegangan yang dihasilkan rangkaian tersebut. Pada pengukuran ini digunakan voltmeter yang memiliki range pengukuran sampai skala milivolt.

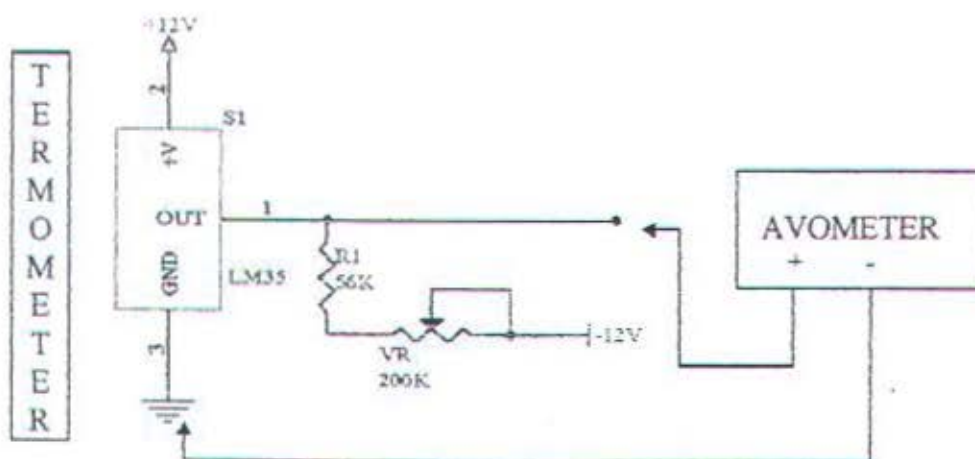
5.1.1. Rangkaian Sensor Suhu IC LM 35

Pada pengukuran suhu dengan rangkaian IC LM35 ini diperoleh hasil pengukuran tegangan terhadap suhu dalam bentuk tabel berikut ini :

Tabel 5.1 Pengukuran Tegangan terhadap Suhu pada Sensor IC LM35

| Tegangan (mV) | Suhu (°C) |
|---------------|-----------|
| 260,9 | 29 |
| 258,9 | 29 |
| 259,5 | 29 |
| 270,6 | 30 |
| 271 | 30 |
| 286,6 | 32 |
| 300,6 | 32 |
| 305 | 32 |
| 307 | 32 |
| 310 | 32 |

Gambar rangkaian pengukuran tegangan dan suhu dengan sensor IC LM35 adalah sebagai berikut :



Gambar 5.1. Rangkaian Pengujian Pengukuran Tegangan & Suhu
Pada Sensor IC LM35

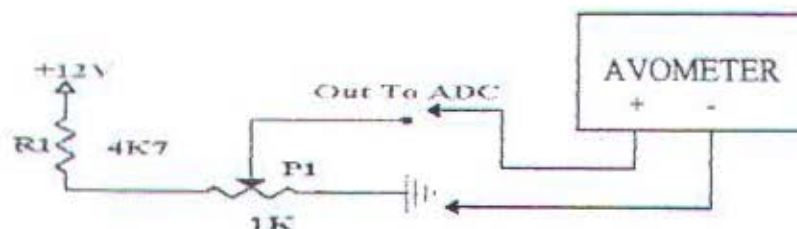
5.1.2. Rangkaian Pembagi Tegangan Potensiometer

Pada pengukuran rangkaian pembagi tegangan dengan potensiometer diperoleh hasil pengukuran tegangan dan harga resistansi dalam bentuk tabel berikut ini:

Tabel 5.2 Pengukuran Tegangan Pada Pembagi Tegangan

| Harga Resistansi (Ω) | Tegangan Output (mV) |
|-------------------------------|----------------------|
| 10,9 | 20,7 |
| 83,7 | 181,2 |
| 121,4 | 265,1 |
| 211,9 | 471 |
| 293 | 667 |
| 437 | 1028 |
| 537 | 1286 |
| 587 | 1427 |
| 654 | 1613 |
| 783 | 2001 |

Gambar rangkaian pengukuran tegangan dan resistansi pada rangkaian pembagi tegangan dengan potensiometer (singleturn) adalah sebagai berikut :



Gambar 5.2. Rangkaian Pengukuran & Pengujian Tegangan Pada Rangkaian Pembagi Tegangan Potensiometer

5.2. Pengujian PPI 8255

Pengujian PPI 8255 ini menentukan keberhasilan dalam mengendalikan peralatan sistem. Dalam pengujian PPI card dipasang pada slot komputer dan kemudian diberikan program sederhana untuk mengeluarkan data pada port-port PPI 8255. Cara kerja program tersebut adalah sebagai berikut :

1. Memberikan control word register untuk inisialisasi PPI 8255 dengan mode 0 dimana PPI berfungsi sebagai input dan output, serta memberikan control word 98H yang mengaktifkan port A sebagai input, port B sebagai output, port C upper sebagai input, dan port C lower sebagai output.
2. Mengaktifkan masing-masing port sesuai dengan fungsinya masing-masing dengan software Turbo Pascal sebagai berikut :

```
Uses Crt,Dos;  
Const  
  A=$300;  
  B=$301;  
  C=$302;  
  CW=$303;  
Begin  
  Repeat  
    Port[CW]:=$98;  
    Data :=Port[A];  
    Port[B]:=$FF;  
    Port[C]:=$0F;
```

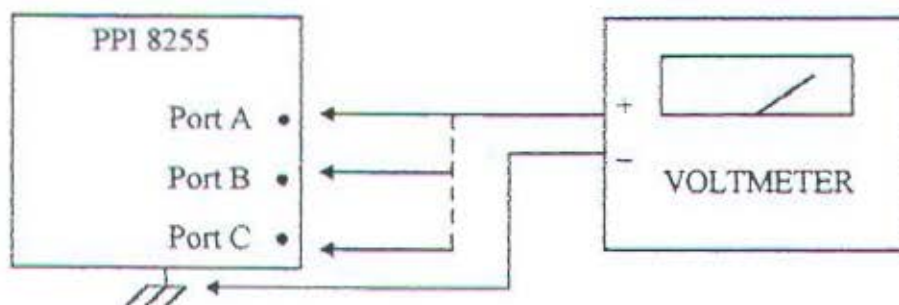


```
EOC:=Port[C]; EOC:= EOC AND $F0;  
Writeln('Input Port A: ',Data,' Input Port C Upper:', EOC);  
Readln;  
Until Keypressd;  
End.
```

Pada software pengujian ini akan ditampilkan data input dari Port A dan Port C Upper ke dalam layar monitor. Penginputan data ke Port A dan Port C Upper dilakukan dengan bantuan dip switch yang dihubungkan ke ground. Untuk Port B dan Port C Lower data outputnya cukup dibaca dengan nyalanya LED yang terpasang pada masing-masing Port.

3. Selain penggunaan LED dalam pengujian ini juga dilakukan pengukuran dengan alat bantu voltmeter untuk mengukur ada tidaknya tegangan dari masing-masing port.
4. Dari hasil pengukuran tersebut akan dibuktikan bahwa PPI 8255 telah bekerja sebagaimana mestinya atau tidak.

Hasil pengukuran tegangan output / input pada PPI 8255 ditunjukkan pada tabel 5.3.



Gambar 5.3. Rangkaian Pengujian Tegangan PPI 8255

Tabel 5.3. Pengukuran Tegangan Output PPI 8255

| Nama Port | Logika '0' (dalam mV) | Logika '1' (dalam V) |
|-----------|-----------------------|----------------------|
| A0 | 40,4 | 4,17 |
| A1 | 40,4 | 4,17 |
| A2 | 40,4 | 4,17 |
| A3 | 40,4 | 4,17 |
| A4 | 40,4 | 4,17 |
| A5 | 40,4 | 4,17 |
| A6 | 40,4 | 4,17 |
| A7 | 40,4 | 4,17 |
| B0 | 40,2 | 4,17 |
| B1 | 40,2 | 4,17 |
| B2 | 40,2 | 4,17 |
| B3 | 40,2 | 4,17 |
| B4 | 40,2 | 4,17 |
| B5 | 40,2 | 4,17 |
| B6 | 40,2 | 4,17 |
| B7 | 40,2 | 4,17 |
| C0 | 40 | 4,12 |
| C1 | 40 | 4,12 |
| C2 | 40 | 4,12 |
| C3 | 40 | 4,12 |
| C4 | 40 | 4,12 |
| C5 | 40 | 4,12 |
| C6 | 40 | 4,12 |
| C7 | 40 | 4,12 |

5.3. Pengujian ADC 0804

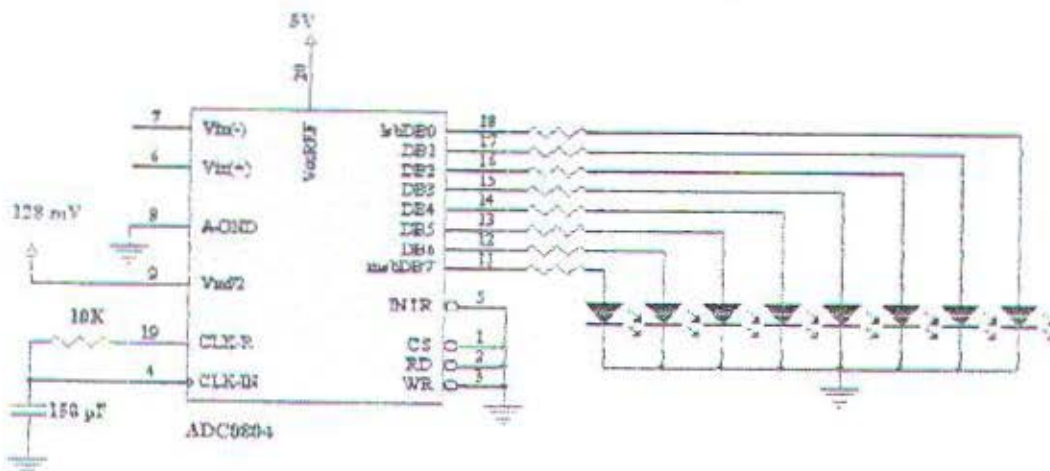
Dalam pengujian ADC ini diperlukan beberapa komponen pembantu antara lain : led, resistor, kapasitor dan power supply.

Dalam pengujian ini untuk menghasilkan tingkat ketelitian sampai dengan 1mV maka $V_{ref}/2$ (pin 9) dapat di supply tegangan DC sebesar 128mV sedangkan V_{cc} disupply tegangan sebesar 5V. Untuk $V_I(-)$ pemberian tegangan sumbernya menggunakan aturan seperti pada tabel dibawah :

Tabel 5.4. Pegukuran Tegangan input ADC

| Range Tegangan Terukur $V_I(+)$ (dalam mV) | $V_I(-)$ (dalam mV) |
|---|------------------------|
| 0 - 254 | 0 |
| 255 - 509 | 255 |
| 510 - 764 | 510 |
| 765 - 1019 | 765 |
| 1020 - 1274 | 1020 |
| 1275 - 1529 | 1275 |
| 1530 - 1784 | 1530 |
| 1785 - 2039 | 1785 |

Gambar rangkaian pengujian ADC 0804 ditunjukkan pada gambar 5.4.



Gambar 5.4. Rangkaian Pengujian ADC

BAB VI

PENUTUP

6.1. Kesimpulan

Dari pembahasan, pengukuran dan pengujian peralatan yang telah dibuat maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Penggunaan komputer dalam proses pengolahan dan pembacaan sinyal input analog memudahkan manusia dalam menganalisa data secara efisien terutama bila sensor sebagai pengirim sinyal input letaknya tidak dalam satu lokasi dengan komputer.
2. Dengan adanya rangkaian akusisi data tersebut dapat dibaca dan dianalisa sinyal-sinyal analog yang diinputkan dengan hasil sinyal yang sebanding dengan sinyal input. Hal ini dikarenakan adanya tingkat ketelitian yang tinggi dalam sistem tersebut.
3. Tersedianya beberapa chanel input dalam rangkaian akusisi data tersebut menyebabkan sistem ini bersifat fleksibel.
4. Pemanfaatan ADC dengan jangkauan ukur (range) input yang dapat diatur tanpa mengurangi ketelitian merupakan keunggulan tersendiri dalam rangkaian tersebut.
5. Proses penyamplingan data dalam sistem akusisi data tersebut menggunakan bantuan perangkat lunak sekaligus berfungsi untuk menampilkan hasil pengolahan data.

6. Dalam pembuatan rangkaian akuisisi data ini dibutuhkan kecermatan dan ketelitian agar sinyal error yang dihasilkan sangat kecil.

2. Saran

Pada sistem akuisisi data ini walaupun memiliki beberapa chanel input yang masing-masing chanel dapat berisi output dari berbagai sensor dengan jangkauan ukur 0 - 2 V, namun dalam tugas akhir ini hanya digunakan 2 chanel input yang masing-masing diaplikasikan dalam bentuk sebuah sensor suhu yaitu sensor IC LM 35 dan sebuah rangkaian pembagi tegangan. Pada pengembangan selanjutnya diharapkan keseluruhan chanel input dapat diaplikasikan dalam berbagai sensor atau rangkaian input sehingga dapat memiliki daya guna yang lebih baik lagi.

Rangkaian akuisisi data dengan 8 channel input ini dalam realisasinya diharapkan dapat dimanfaatkan untuk pembacaan data yang membutuhkan input lebih dari satu seperti misalnya dikembangkan pada Badan Meteorologi dan Geofisika untuk pengukuran beberapa parameter cuaca seperti pengukuran suhu, kelembaban, kecepatan angin dan curah hujan.

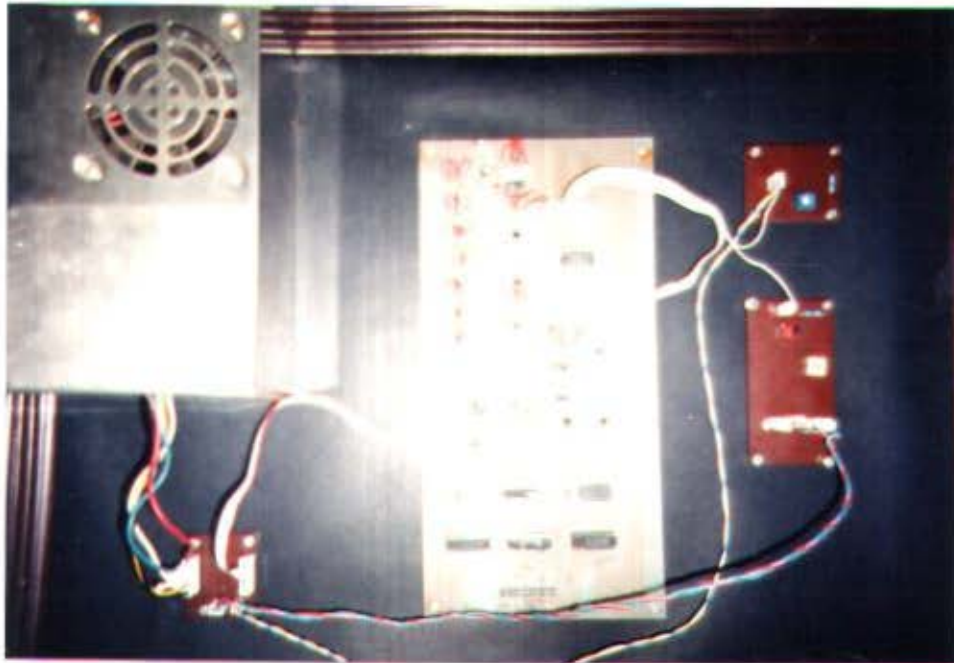
DAFTAR PUSTAKA

1. Coughlin, Robert F., Fredrick F., Penguat Operational dan Rangkaian Terpadu Linier, Herman, Ir., Alih Bahasa, Erlangga, Jakarta, 1995.
2. Den Heijer, PC., Tolsma, R., Komunikasi Data, PT Elex Media Komputindo, Jakarta. 1988.
3. Eggebrecht, Lewis C., Interfacing to IBM PC, Howard W. Sons and Co., Inc, Indiana Polis USA, 1985.
4. Fairchild, TTL Data Book, Fairchild Camera and Instrument Cooperation, California, 1978.
5. Jerald G. Graeme, Aplication of Operational Amplifier, Mc Graw Hill International Book.
6. Maloney, Timothy, Industrial Solid State Electronics Device and System, Prentice, Hall, Inc., Englewood Cliffs, New Jersey 0763.
7. Steeman, JPM, Data Sheet Book 2, PT. Elex Media Komputindo, Jakarta, 1988.
8. Wasito S., Alih Bahasa, Data Sheet Book 1, PT. Elex Media Komputindo, Gramedia, jakarta.
9. _____, National Semiconductor, Linear Data Book, Santa Clara, California.

LAMPIRAN A

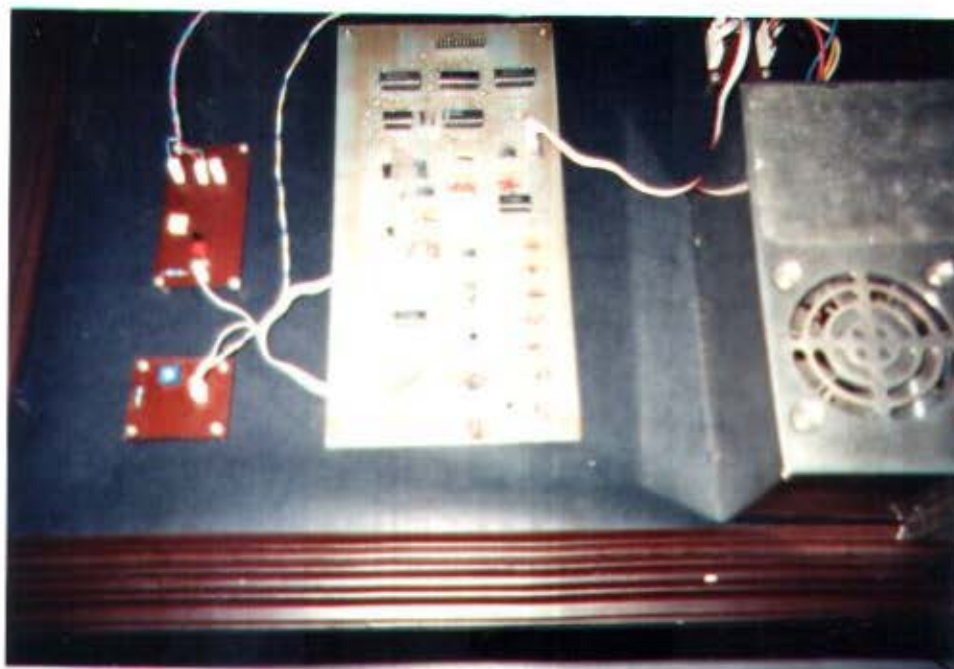
GAMBAR RANGKAIAN TUGAS AKHIR

AKUSISI DATA 8 CHANNEL INPUT



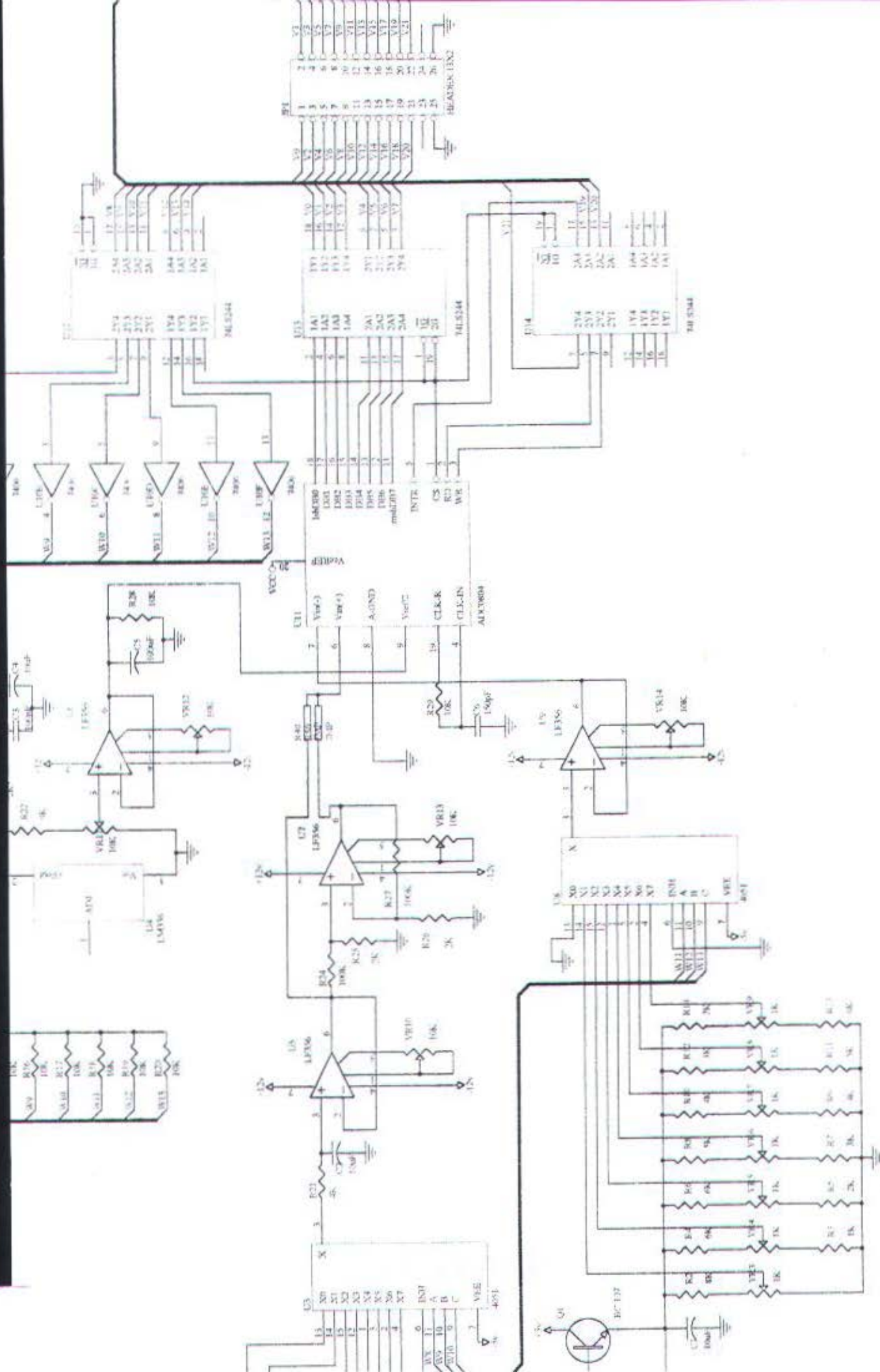
Gambar Rangkaian Tugas Akhir

Akusisi Data 8 Channel Input



Gambar Rangkaian Tugas Akhir

Akusisi Data Dilihat Dari Sisi Lain



LAMPIRAN B

LISTING PROGRAM AKUSISI DATA

LISTING PROGRAM AKUSISI DATA

```
USES DOS,CRT,TOOL;
CONST
A = $300; B = $301; C = $302; CW= $303;
VAR
DATA,TEST,I,SELECT,
PILIH,RANGE,EOC,VREFF :INTEGER;
OUTPUT,SUHU           :REAL ;
KATA                   :STRING;
TEKAN                  :CHAR;

PROCEDURE TABEL(KATA:STRING);
BEGIN
FRAME(25,2,50,25,YELLOW,LIGHTBLUE,'2',FALSE);
GOTOXY(30,3);WRITE(KATA);
GOTOXY(25,4);WRITE('LMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM9');
END;

PROCEDURE REFF1;
BEGIN
IF VREFF=$3F THEN RANGE:=0      ELSE
IF VREFF=$37 THEN RANGE:=256    ELSE
IF VREFF=$2F THEN RANGE:=512    ELSE
IF VREFF=$27 THEN RANGE:=768    ELSE
IF VREFF=$1F THEN RANGE:=1024   ELSE
IF VREFF=$17 THEN RANGE:=1280   ELSE
IF VREFF=$0F THEN RANGE:=1536   ELSE
IF VREFF=$07 THEN RANGE:=1792
END;

PROCEDURE REFF2;
BEGIN
IF VREFF=$3E THEN RANGE:=0      ELSE
IF VREFF=$36 THEN RANGE:=256    ELSE
```



```
IF VREFF=$2E THEN RANGE:=512 ELSE
IF VREFF=$26 THEN RANGE:=768 ELSE
IF VREFF=$1E THEN RANGE:=1024 ELSE
IF VREFF=$16 THEN RANGE:=1280 ELSE
IF VREFF=$0E THEN RANGE:=1536 ELSE
IF VREFF=$06 THEN RANGE:=1792
```

```
END;
```

```
PROCEDURE REFF3;
```

```
BEGIN
```

```
IF VREFF=$3D THEN RANGE:=0 ELSE
IF VREFF=$35 THEN RANGE:=256 ELSE
IF VREFF=$2D THEN RANGE:=512 ELSE
IF VREFF=$25 THEN RANGE:=768 ELSE
IF VREFF=$1D THEN RANGE:=1024 ELSE
IF VREFF=$15 THEN RANGE:=1280 ELSE
IF VREFF=$0D THEN RANGE:=1536 ELSE
IF VREFF=$05 THEN RANGE:=1792
```

```
END;
```

```
PROCEDURE REFF4;
```

```
BEGIN
```

```
IF VREFF=$3C THEN RANGE:=0 ELSE
IF VREFF=$34 THEN RANGE:=256 ELSE
IF VREFF=$2C THEN RANGE:=512 ELSE
IF VREFF=$24 THEN RANGE:=768 ELSE
IF VREFF=$1C THEN RANGE:=1024 ELSE
IF VREFF=$14 THEN RANGE:=1280 ELSE
IF VREFF=$0C THEN RANGE:=1536 ELSE
IF VREFF=$04 THEN RANGE:=1792
```

```
END;
```

```
PROCEDURE REFF5;
```

```
BEGIN
```

```
IF VREFF=$3B THEN RANGE:=0 ELSE
IF VREFF=$33 THEN RANGE:=256 ELSE
```

```
IF VREFF=$2B THEN RANGE:=512 ELSE
IF VREFF=$23 THEN RANGE:=768 ELSE
IF VREFF=$1B THEN RANGE:=1024 ELSE
IF VREFF=$13 THEN RANGE:=1280 ELSE
IF VREFF=$0B THEN RANGE:=1536 ELSE
IF VREFF=$03 THEN RANGE:=1792
```

```
END;
```

```
PROCEDURE REFF6;
```

```
BEGIN
```

```
IF VREFF=$3A THEN RANGE:=0 ELSE
IF VREFF=$32 THEN RANGE:=256 ELSE
IF VREFF=$2A THEN RANGE:=512 ELSE
IF VREFF=$22 THEN RANGE:=768 ELSE
IF VREFF=$1A THEN RANGE:=1024 ELSE
IF VREFF=$12 THEN RANGE:=1280 ELSE
IF VREFF=$0A THEN RANGE:=1536 ELSE
IF VREFF=$02 THEN RANGE:=1792
```

```
END;
```

```
PROCEDURE REFF7;
```

```
BEGIN
```

```
IF VREFF=$39 THEN RANGE:=0 ELSE
IF VREFF=$31 THEN RANGE:=256 ELSE
IF VREFF=$29 THEN RANGE:=512 ELSE
IF VREFF=$21 THEN RANGE:=768 ELSE
IF VREFF=$19 THEN RANGE:=1024 ELSE
IF VREFF=$11 THEN RANGE:=1280 ELSE
IF VREFF=$09 THEN RANGE:=1536 ELSE
IF VREFF=$01 THEN RANGE:=1792
```

```
END;
```

```
PROCEDURE REFF8;
```

```
BEGIN
```

```
IF VREFF=$38 THEN RANGE:=0 ELSE
IF VREFF=$30 THEN RANGE:=256 ELSE
```

```
IF VREFF=$28 THEN RANGE:=512 ELSE
IF VREFF=$20 THEN RANGE:=768 ELSE
IF VREFF=$18 THEN RANGE:=1024 ELSE
IF VREFF=$10 THEN RANGE:=1280 ELSE
IF VREFF=$08 THEN RANGE:=1536 ELSE
IF VREFF=$00 THEN RANGE:=1792
```

```
END;
```

```
PROCEDURE CHANEL1;
```

```
BEGIN
```

```
PORT[B] := VREFF;
```

```
PORT[C] := $07;
```

```
PORT[C] := $00;
```

```
PORT[C] := $07;
```

```
REPEAT
```

```
    EOC := PORT[C];
```

```
    TEST:= EOC AND $10;
```

```
UNTIL TEST<>$10;
```

```
REFF1;
```

```
DATA := PORT[A];DELAY(10);
```

```
END;
```

```
PROCEDURE CHANEL2;
```

```
BEGIN
```

```
PORT[B] := VREFF;
```

```
PORT[C] := $07;
```

```
PORT[C] := $00;
```

```
PORT[C] := $07;
```

```
REPEAT
```

```
    EOC := PORT[C];
```

```
    TEST:= EOC AND $10;
```

```
UNTIL TEST<>$10;
```

```
REFF2;
```

```
DATA := PORT[A]; DELAY(10);
```

```
END;
```



```
PROCEDURE CHANEL3;
BEGIN
PORT[B] := VREFF;
PORT[C] := $07;
PORT[C] := $00;
PORT[C] := $07;
REPEAT
  EOC := PORT[C];
  TEST:= EOC AND $10;
UNTIL TEST<>$10;
REFF3;
DATA := PORT[A]; DELAY(10);
END;

PROCEDURE CHANEL4;
BEGIN
PORT[B] := VREFF;
PORT[C] := $07;
PORT[C] := $00;
PORT[C] := $07;
REPEAT
  EOC := PORT[C];
  TEST:= EOC AND $10;
UNTIL TEST<>$10;
REFF4;
DATA := PORT[A]; DELAY(10);
END;

PROCEDURE CHANEL5;
BEGIN
PORT[B] := VREFF;
PORT[C] := $07;
PORT[C] := $00;
PORT[C] := $07;
```

```
REPEAT
    EOC := PORT[C];
    TEST:= EOC AND $10;
UNTIL TEST<>$10;
REFF5;
DATA := PORT[A]; DELAY(10);
END;

PROCEDURE CHANEL6;
BEGIN
PORT[B] := VREFF;
PORT[C] := $07;
PORT[C] := $00;
PORT[C] := $07;
REPEAT
    EOC := PORT[C];
    TEST:= EOC AND $10;
UNTIL TEST<>$10;
REFF6;
DATA := PORT[A]; DELAY(10);
END;

PROCEDURE CHANEL7;
BEGIN
PORT[B] := VREFF;
PORT[C] := $07;
PORT[C] := $00;
PORT[C] := $07;
REPEAT
    EOC := PORT[C];
    TEST:= EOC AND $10;
UNTIL TEST<>$10;
REFF7;
DATA := PORT[A]; DELAY(10);
END;
```

```
PROCEDURE CHANEL8;
BEGIN
PORT[B] := VREFF;
PORT[C] := $07;
PORT[C] := $00;
PORT[C] := $07;
REPEAT
  EOC := PORT[C];
  TEST:= EOC AND $10;
UNTIL TEST<>$10;
REFF8;
DATA := PORT[A]; DELAY(10);
END;

BEGIN
CLRSCR;
PORT[CW]:=$98;
TEXTBACKGROUND(1);CLRSCR;FRAME(8,1,70,14,14,7,'2',TRUE);
FRAME(22,2,55,6,12+128,11,'2',TRUE);TEXTCOLOR(1);
GOTOXY(25,3);TEXTBACKGROUND(11);
WRITE('      " TUGAS AKHIR "      ');GOTOXY(35,4);
WRITE('O L E H :');GOTOXY(23,5);
WRITE('      RASTONO.H & M.ISLAN ');
FRAME(12,9,66,11,11,12,'2',TRUE);TEXTCOLOR(14+128);
TEXTBACKGROUND(12);GOTOXY(13,10);
WRITE('      " PROGRAM SISTEM AKUSISI DATA BERBASIS IBM-PC "');
TEXTBACKGROUND(1);TEXTCOLOR(14);GOTOXY(23,21);
WRITE('Tekan tombol Enter untuk melanjutkan,');
GOTOXY(29,23);WRITE('atau ESC untuk keluar !');
TEKAN:=READKEY;  IF (TEKAN= #27) THEN
  BEGIN
    TEXTBACKGROUND(0);TEXTCOLOR(15);CLRSCR;
```



```
HALT(1)
END;
CLRSCR; SAVESCREEN(15,5,36,17);
REPEAT
FRAME(26,3,55,12,YELLOW,LIGHTBLUE,'2',TRUE);
INITPROMPT(YELLOW,BLUE,BLACK,GREEN,RED,WHITE,BLUE,33,25,24);
TEXTCOLOR(YELLOW);TEXTBACKGROUND(BLUE);
GOTOXY(28,4);WRITE('    M E N U    U T A M A    ');
GOTOXY(26,5);WRITE('LMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM9');
FRAME(22,23,59,25,YELLOW,BLUE,'2',FALSE);
PROMPT(34,7,'~CHANNEL INPUT');
MESSAGE(' TUGAS AKHIR SISTEM AKUSISI DATA ');
PROMPT(38,9,'E~XIT');
MESSAGE(' TUGAS AKHIR SISTEM AKUSISI DATA ');
SELECT:=GETPROMPT; TEXTBACKGROUND(BLUE);
CASE SELECT OF
1:BEGIN
    FRAME(45,9,67,18,YELLOW,BLACK,'1',TRUE);
    INITPROMPT(WHITE,BLACK,YELLOW,BLUE,RED,WHITE,BLUE,0,0,0);
    PROMPT(47,10,'~1. CHANNEL 1');
    PROMPT(47,11,'~2. CHANNEL 2');
    PROMPT(47,12,'~3. CHANNEL 3');
    PROMPT(47,13,'~4. CHANNEL 4');
    PROMPT(47,14,'~5. CHANNEL 5');
    PROMPT(47,15,'~6. CHANNEL 6');
    PROMPT(47,16,'~7. CHANNEL 7');
    PROMPT(47,17,'~8. CHANNEL 8');
    PILIH:=GETPROMPT;CLRSCR;
    CASE PILIH OF
    1:BEGIN
        I:=6;TABEL('SUHU / TEMPERATUR');
        REPEAT
        VREF:=33F;
```

```
REPEAT
CHANEL1;
IF DATA=255 THEN VREFF:=VREFF-8;
UNTIL (DATA <> 255);
IF DATA<0 THEN
BEGIN
VREFF:=VREFF+8;
CHANEL1;
END;
OUTPUT := DATA+RANGE;
SUHU := OUTPUT/10;
IF I=24 THEN
BEGIN
CLRSCR;I:=6;TABEL('SUHU / TEMPERATUR');
END;
GOTOXY(35,I);WRITE(SUHU:3:1,' xC');DELAY(700);INC(I);
UNTIL KEYPRESSED;
END;
2:BEGIN
I:=6;TABEL('TEGANGAN OUTPUT');
REPEAT
VREFF:=$3E;
REPEAT
CHANEL2;
IF DATA=255 THEN VREFF:=VREFF-8;
UNTIL (DATA <> 255);
IF DATA<0 THEN
BEGIN
VREFF:=VREFF+8;
CHANEL2;
END;
OUTPUT := DATA+RANGE;
```

```
IF I=24 THEN
  BEGIN
    CLRSCR;I:=6;TABEL('TEGANGAN OUTPUT');
    END;
  GOTOXY(35,I);WRITE(OUTPUT:4:0,' mV');DELAY(700);INC(I);
  UNTIL KEYPRESSED;
  END;
3:BEGIN
  I:=6;TABEL('TEGANGAN OUTPUT');
  REPEAT
    VREFF:=$3D;
    REPEAT
      CHANEL3;
      IF DATA=255 THEN VREFF:=VREFF-8;
      UNTIL (DATA <> 255);
      IF DATA<0 THEN
        BEGIN
          VREFF:=VREFF+8;
          CHANEL3;
        END;
      OUTPUT := DATA+RANGE;
      IF I=24 THEN
        BEGIN
          CLRSCR;I:=6;TABEL('TEGANGAN OUTPUT');
          END;
        GOTOXY(35,I);WRITE(OUTPUT:4:0,' mV');DELAY(700);INC(I);
        UNTIL KEYPRESSED;
        END;
4:BEGIN
  I:=6;TABEL('TEGANGAN OUTPUT');
  REPEAT
    VREFF:=$3C;
```



```
REPEAT
CHANEL4;
IF DATA=255 THEN VREFF:=VREFF-8;
UNTIL (DATA <> 255);
IF DATA<0 THEN
BEGIN
VREFF:=VREFF+8;
CHANEL4;
END;
OUTPUT := DATA+RANGE;
IF I=24 THEN
BEGIN
CLRSCR;I:=6;TABEL('TEGANGAN OUTPUT');
END;
GOTOXY(35,I);WRITE(OUTPUT:4:0,' mV');DELAY(700);INC(I);
UNTIL KEYPRESSED;
END;
5:BEGIN
I:=6;TABEL('TEGANGAN OUTPUT');
REPEAT
VREFF:=$3B;
REPEAT
CHANEL5;
IF DATA=255 THEN VREFF:=VREFF-8;
UNTIL (DATA <> 255);
IF DATA<0 THEN
BEGIN
VREFF:=VREFF+8;
CHANEL5;
END;
OUTPUT := DATA+RANGE;
```

```
IF I=24 THEN
  BEGIN
    CLRSCR;I:=6;TABEL('TEGANGAN OUTPUT');
    END;
  GOTOXY(35,I);WRITE(OUTPUT:4:0,' mV');DELAY(700);INC(I);
  UNTIL KEYPRESSED;
  END;
6:BEGIN
  I:=6;TABEL('TEGANGAN OUTPUT');
  REPEAT
    VREFF:=$3A;
    REPEAT
      CHANEL6;
      IF DATA=255 THEN VREFF:=VREFF-8;
    UNTIL (DATA <> 255);
    IF DATA<0 THEN
      BEGIN
        VREFF:=VREFF+8;
        CHANEL6;
      END;
    OUTPUT := DATA+RANGE;
    IF I=24 THEN
      BEGIN
        CLRSCR;I:=6;TABEL('TEGANGAN OUTPUT');
        END;
      GOTOXY(35,I);WRITE(OUTPUT:4:0,' mV');DELAY(700);INC(I);
      UNTIL KEYPRESSED;
      END;
7:BEGIN
  I:=6;TABEL('TEGANGAN OUTPUT');
  REPEAT
    VREFF:=$39;
```

```
REPEAT
CHANEL7;
IF DATA=255 THEN VREFF:=VREFF-8;
UNTIL (DATA <> 255);
IF DATA<0 THEN
BEGIN
VREFF:=VREFF+8;
CHANEL7;
END;
OUTPUT := DATA+RANGE;
IF I=24 THEN
BEGIN
CLRSCR;I:=6;TABEL('TEGANGAN OUTPUT');
END;
GOTOXY(35,1);WRITE(OUTPUT:4:0,' mV');DELAY(700);INC(I);
UNTIL KEYPRESSED;
END;
8:BEGIN
I:=6;TABEL('TEGANGAN OUTPUT');
REPEAT
VREFF:=038;
REPEAT
CHANEL8;
IF DATA=255 THEN VREFF:=VREFF-8;
UNTIL (DATA <> 255);
IF DATA<0 THEN
BEGIN
VREFF:=VREFF+8;
CHANEL8;
END;
OUTPUT := DATA+RANGE;
```

```
IF I=24 THEN
  BEGIN
    CLRSCR;I:=6;TABEL('TEGANGAN OUTPUT');
    END;
  GOTOXY(35,1);WRITE(OUTPUT:4:0,' mV');DELAY(700);INC(I);
  UNTIL KEYPRESSED;
  END;
END;RESTOREPROMPT;WINDOW(1,1,80,25);CLRSCR;
END;
2:BEGIN
  CLRSCR;EXIT;
  END;
END;
RESTOREPROMPT;
UNTIL SELECT=2;
RESTORESCREEN;TEXTBACKGROUND(BLACK);CLRSCR;
END.
```


LAMPIRAN C

DATA SHEET BOOK

8255 Programmable Peripheral Interface

- MCS-85™ Compatible 8255A-5
- 24 Programmable I/O Pins
- Completely TTL Compatible
- Fully Compatible with Intel Microprocessor Families
- Improved Timing Characteristics
- Direct Bit Set/Reset Capability Easing Control Application Interface
- Reduces System Package Count
- Improved DC Driving Capability
- Available in EXPRESS
 - Standard Temperature Range
 - Extended Temperature Range
- 40 Pin DIP Package or 44 Lead PLCC

The Intel 8255A is a general purpose programmable I/O device designed for use with Intel microprocessors. It has 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. In the first mode (MODE 0), each group of 12 I/O pins may be programmed in sets of 4 to be input or output. In MODE 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals. The third mode of operation (MODE 2) is a bidirectional bus mode which uses 8 lines for a bidirectional bus, and 5 lines, borrowing one from the other group, for handshaking.

8255A FUNCTIONAL DESCRIPTION

General

The 8255A is a programmable peripheral interface (PPI) device designed for use in Intel microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 8255A is programmed by the system

software so that normally no external logic is necessary to interface peripheral devices or structures.

Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to interface the 8255A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

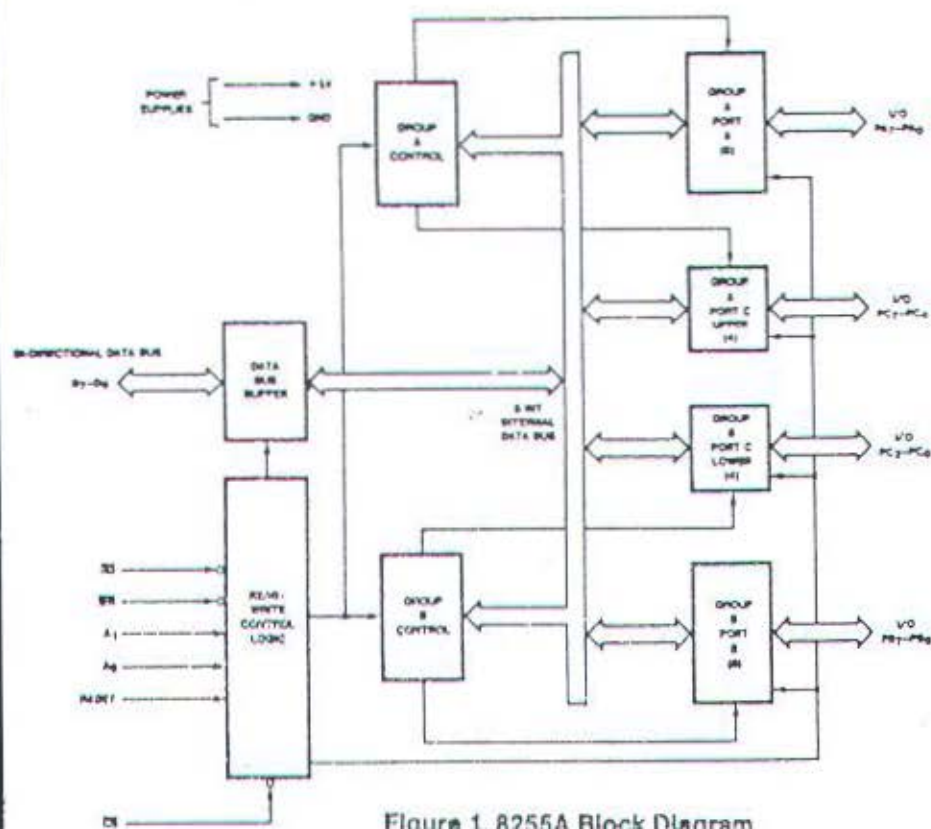


Figure 1. 8255A Block Diagram



Figure 2. Pin Configuration

Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

(CS)

Chip Select. A "low" on this input pin enables the communication between the 8255A and the CPU.

(RD)

Read. A "low" on this input pin enables the 8255A to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to "read from" the 8255A.

(WR)

Write. A "low" on this input pin enables the CPU to write data or control words into the 8255A.

(A₀ and A₁)

Port Select 0 and Port Select 1. These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the

control word registers. They are normally connected to the least significant bits of the address bus (A₀ and A₁).

8255A BASIC OPERATION

| A ₁ | A ₀ | \overline{RD} | \overline{WR} | \overline{CS} | Input Operation (READ) |
|----------------|----------------|-----------------|-----------------|-----------------|---------------------------------|
| 0 | 0 | 0 | 1 | 0 | Port A → Data Bus |
| 0 | 1 | 0 | 1 | 0 | Port B → Data Bus |
| 1 | 0 | 0 | 1 | 0 | Port C → Data Bus |
| | | | | | Output Operation (WRITE) |
| 0 | 0 | 1 | 0 | 0 | Data Bus → Port A |
| 0 | 1 | 1 | 0 | 0 | Data Bus → Port B |
| 1 | 0 | 1 | 0 | 0 | Data Bus → Port C |
| 1 | 1 | 1 | 0 | 0 | Data Bus → Control |
| | | | | | Disable Function |
| X | X | X | X | 1 | Data Bus → 3-State |
| 1 | 1 | 0 | 1 | 0 | Illegal Condition |
| X | X | 1 | 1 | 0 | Data Bus → 3-State |

(RESET)

Reset. A "high" on this input clears the control register and all ports (A, B, C) are set to the input.

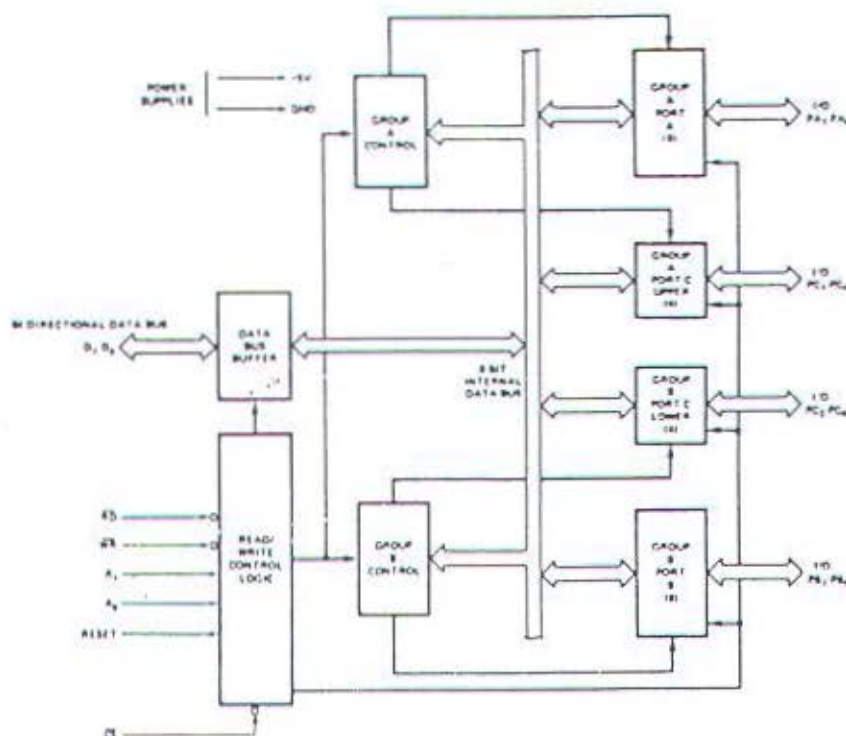


Figure 3. 8255A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions

8255 Programmable Peripheral Interface

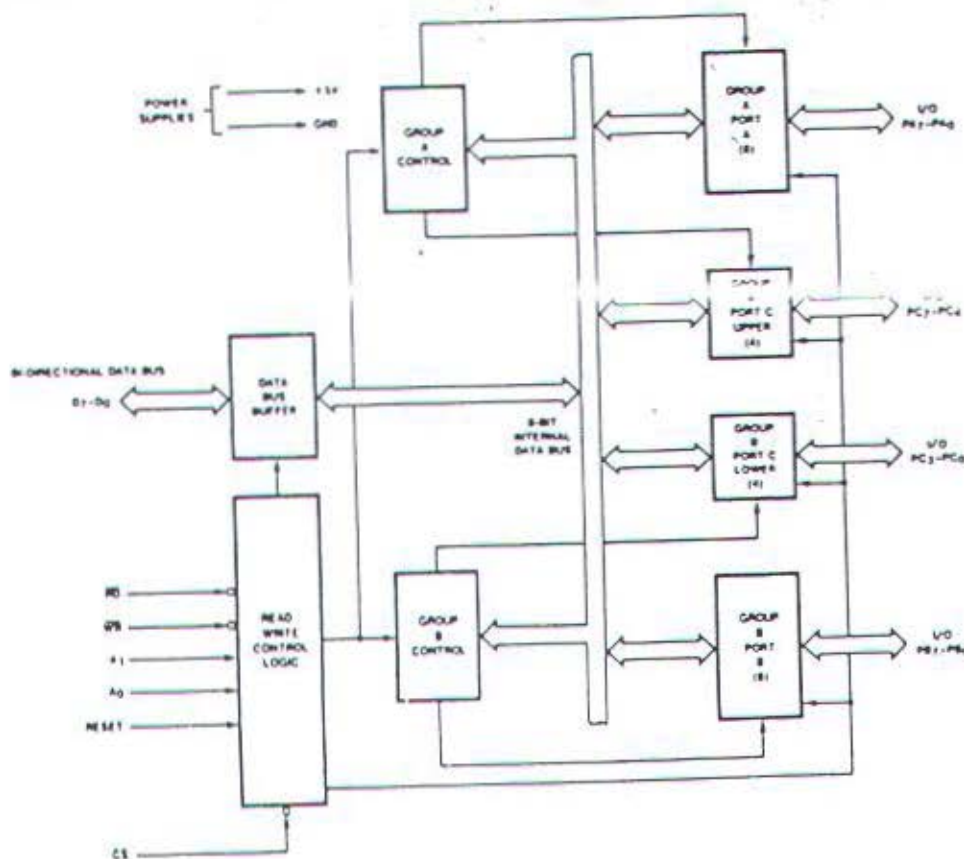


Figure 4. 8255A Block Diagram Showing Group A and Group B Control Functions

Group A and Group B Controls

The functional configuration of each port is programmed by the systems software. In essence, the CPU "outputs" a control word to the 8255A. The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 8255A.

Each of the Control blocks (Group A and Group B) accepts "commands" from the Read/Write Control Logic, receives "control words" from the internal data bus and issues the proper commands to its associated ports.

Control Group A—Port A and Port C upper (C7-C4)

Control Group B—Port B and Port C lower (C3-C0)

The Control Word Register can Only be written into. No Read operation of the Control Word Register is allowed.

Ports A, B, and C

The 8255A contains three 8-bit ports (A, B, and C). All can be configured in a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 8255A.

Pin Configuration



Port A. One 8-bit data output latch/buffer and one 8-bit data input latch.

Port B. One 8-bit data input/output latch/buffer and one 8-bit data input buffer.

Port C. One 8-bit data output latch/buffer and one

Programmable Peripheral Interface 8255

8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B.

Pin Names

| D ₇ -D ₀ | Data Bus (8-bit) |
|----------------------------------|------------------|
| RESET | Reset Input |
| \overline{CS} | Chip Select |
| \overline{RD} | Read Input |
| \overline{WR} | Write Input |
| A ₀ , A ₁ | Port Address |
| PA ₇ -PA ₀ | Port A (BIT) |
| PB ₇ -PB ₀ | Port B (BIT) |
| PC ₇ -PC ₀ | Port C (BIT) |
| V _{CC} | + 5 Volts |
| GND | 0 Volts |

8255A OPERATIONAL DESCRIPTION

Mode Selection

There are three basic modes of operation that can be selected by the system software:

Mode 0—Basic Input/Output

Mode 1—Strobed Input/Output

Mode 2—Bi-Directional Bus

When the reset input goes "high" all ports will be set to the input mode (i.e., all 24 lines will be in the high impedance state). After the reset is removed the 8255A can remain in the input mode with no additional initialization required. During the execution of the system program any of the other modes may be selected using a single output instruction. This allows a single 8255A to service a variety of peripheral devices with a simple software maintenance routine.

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance; Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results, Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an

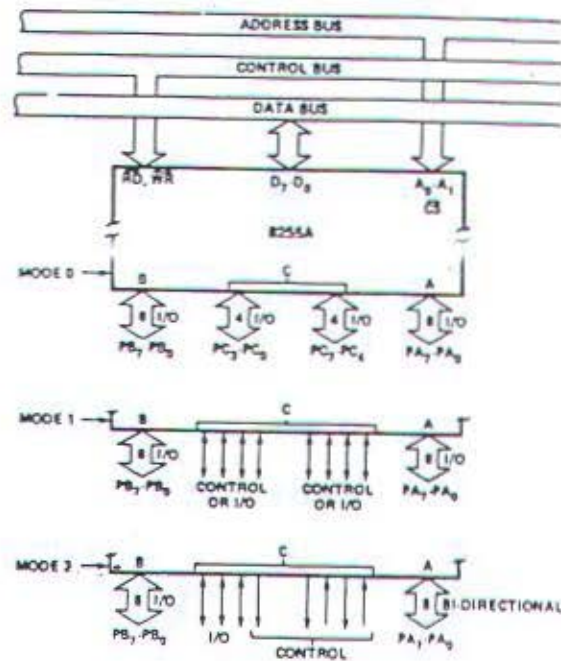
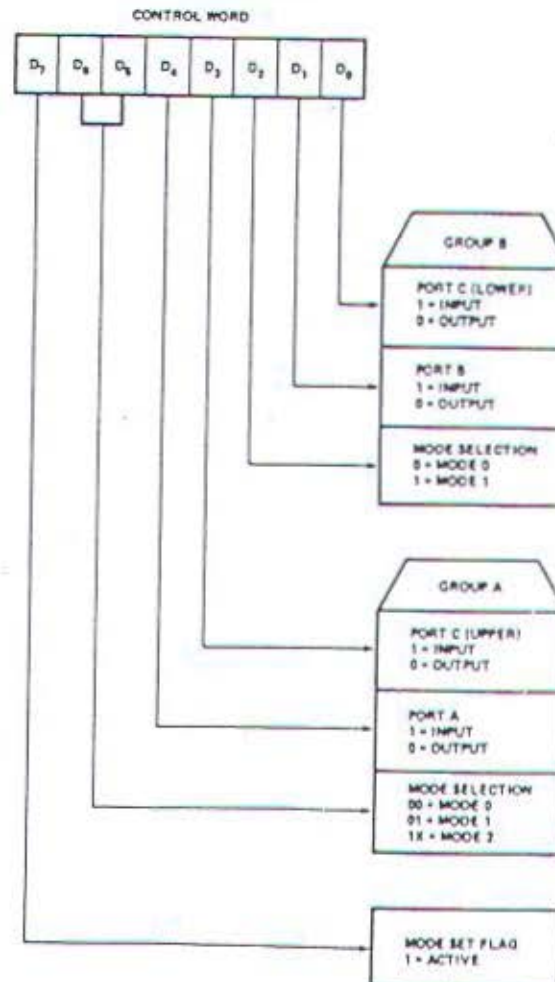


Figure 5. Basic Mode Definitions and Bus Interface



8255 Programmable Peripheral Interface

The mode definitions and possible mode combinations may seem confusing at first but after a cursory review of the complete device operation a simple, logical I/O approach will surface. The design of the 8255A has taken into account things such as efficient PC board layout, control signal definition vs PC layout and complete functional flexibility to support almost any peripheral device with no external logic. Such design represents the maximum use of the available pins.

Single Bit Set/Reset Feature

Any of the eight bits of Port C can be Set or Reset using a single OUTput instruction. This feature reduces software requirements in Control-based applications.

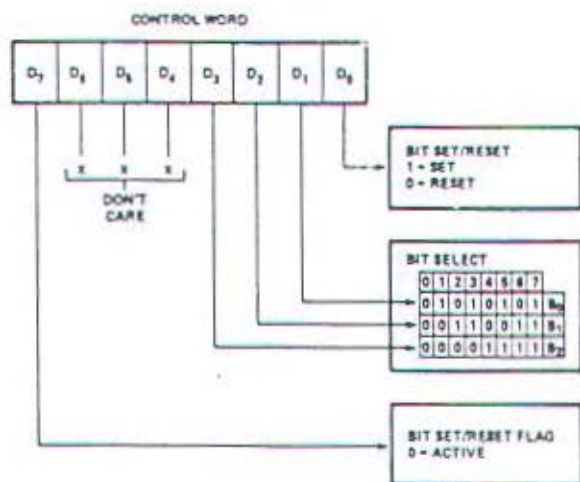


Figure 7. Bit Set/Reset Format

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were data output ports.

MODE 0 (BASIC INPUT)



Interrupt Control Functions

When the 8255A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip-flop using the bit set/reset function of port C.

This function allows the Programmer to disallow or allow a specific I/O device to interrupt the CPU without affecting any other device in the interrupt structure.

INTE flip-flop definition:

(BIT-SET)—INTE is set—Interrupt enable

(BIT-RESET)—INTE is RESET—Interrupt disable

NOTE:

All Mask flip-flops are automatically reset during mode selection and device Reset.

Operating Modes

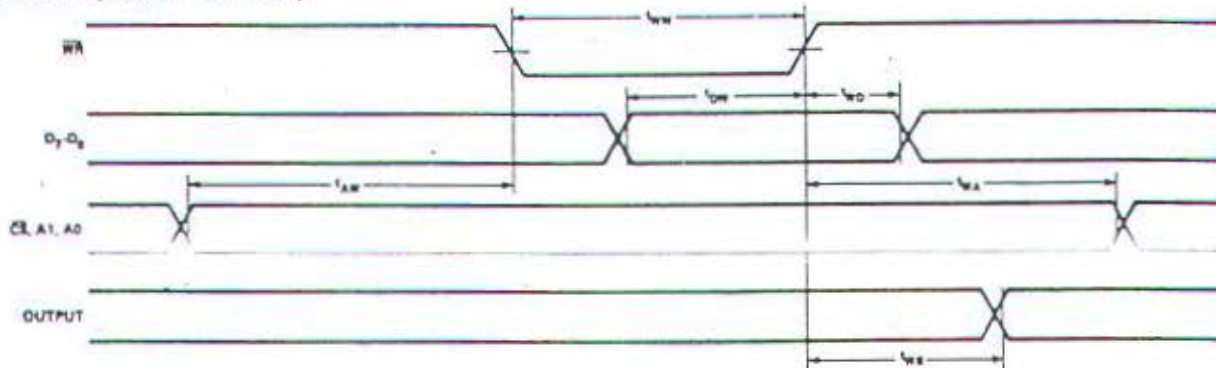
MODE 0 (Basic Input/Output). This functional configuration provides simple input and output operations for each of the three ports. No "handshaking" is required, data is simply written to or read from a specified port.

Mode 0 Basic Functional Definitions:

- Two 8-bit ports and two 4-bit ports.
- Any port can be input or output.
- Outputs are latched.
- Inputs are not latched.
- 16 different Input/Output configurations are possible in this Mode.

Programmable Peripheral Interface 8255

MODE 0 (BASIC OUTPUT)



MODE 0 PORT DEFINITION

| A | | B | | Group A | | | Group B | |
|----------------|----------------|----------------|----------------|---------|----------------|----|---------|----------------|
| D ₄ | D ₃ | D ₁ | D ₀ | Port A | Port C (Upper) | # | Port B | Port C (Lower) |
| 0 | 0 | 0 | 0 | OUTPUT | OUTPUT | 0 | OUTPUT | OUTPUT |
| 0 | 0 | 0 | 1 | OUTPUT | OUTPUT | 1 | OUTPUT | INPUT |
| 0 | 0 | 1 | 0 | OUTPUT | OUTPUT | 2 | INPUT | OUTPUT |
| 0 | 0 | 1 | 1 | OUTPUT | OUTPUT | 3 | INPUT | INPUT |
| 0 | 1 | 0 | 0 | OUTPUT | INPUT | 4 | OUTPUT | OUTPUT |
| 0 | 1 | 0 | 1 | OUTPUT | INPUT | 5 | OUTPUT | INPUT |
| 0 | 1 | 1 | 0 | OUTPUT | INPUT | 6 | INPUT | OUTPUT |
| 0 | 1 | 1 | 1 | OUTPUT | INPUT | 7 | INPUT | INPUT |
| 1 | 0 | 0 | 0 | INPUT | OUTPUT | 8 | OUTPUT | OUTPUT |
| 1 | 0 | 0 | 1 | INPUT | OUTPUT | 9 | OUTPUT | INPUT |
| 1 | 0 | 1 | 0 | INPUT | OUTPUT | 10 | INPUT | OUTPUT |
| 1 | 0 | 1 | 1 | INPUT | OUTPUT | 11 | INPUT | INPUT |
| 1 | 1 | 0 | 0 | INPUT | INPUT | 12 | OUTPUT | OUTPUT |
| 1 | 1 | 0 | 1 | INPUT | INPUT | 13 | OUTPUT | INPUT |
| 1 | 1 | 1 | 0 | INPUT | INPUT | 14 | INPUT | OUTPUT |
| 1 | 1 | 1 | 1 | INPUT | INPUT | 15 | INPUT | INPUT |

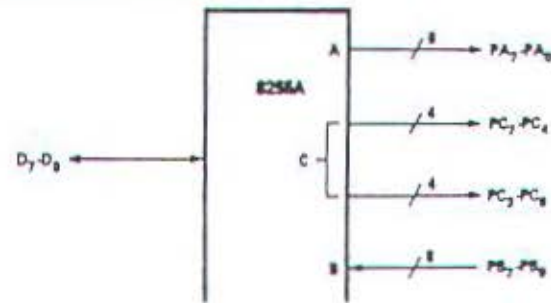
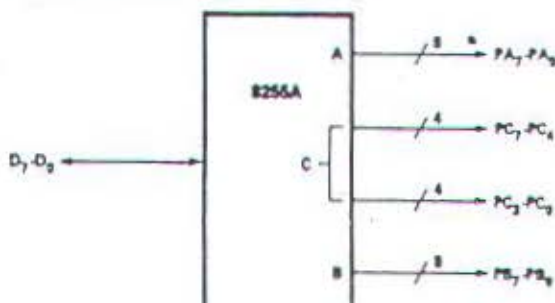
MODE CONFIGURATIONS

CONTROL WORD #0

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

CONTROL WORD #2

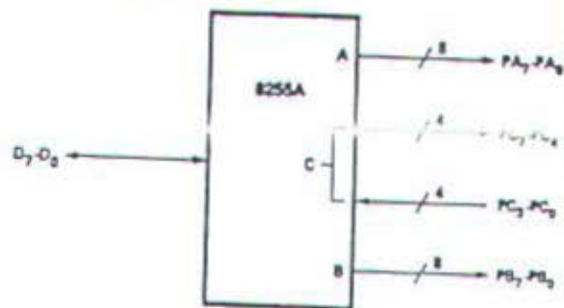
| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |



8255 Programmable Peripheral Interface

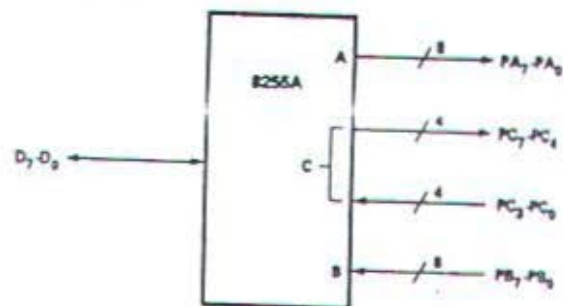
CONTROL WORD #1

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |



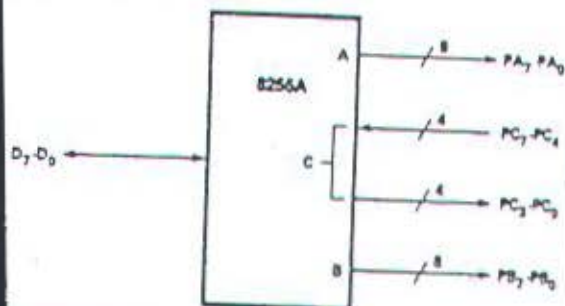
CONTROL WORD #3

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |



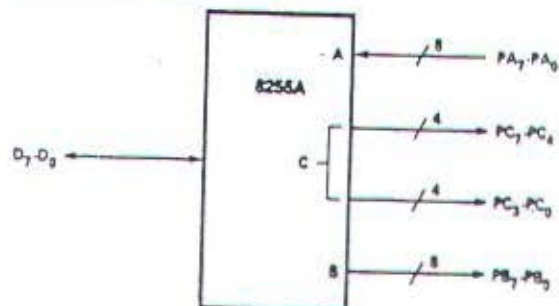
CONTROL WORD #4

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |



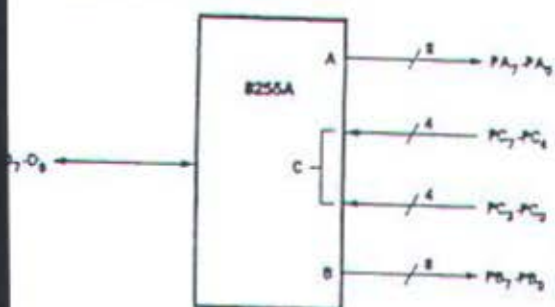
CONTROL WORD #6

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |



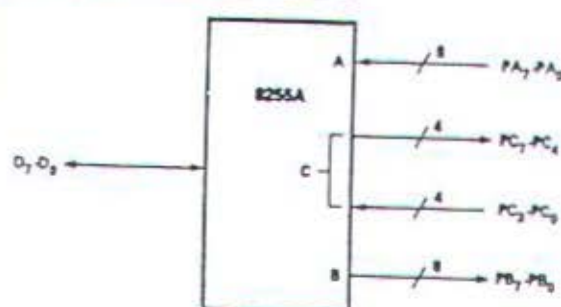
CONTROL WORD #5

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |



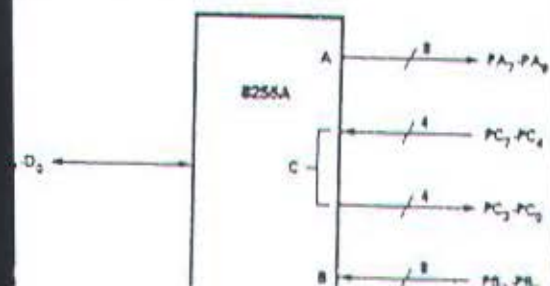
CONTROL WORD #9

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |



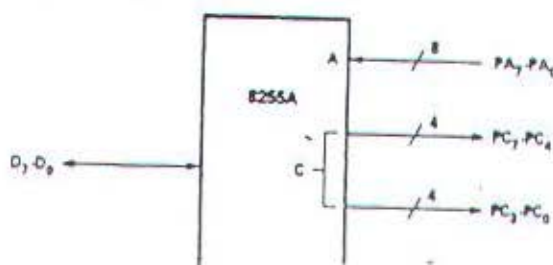
CONTROL WORD #8

| D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 0 | 0 | 0 | 1 | 0 | 1 | 0 |



CONTROL WORD #10

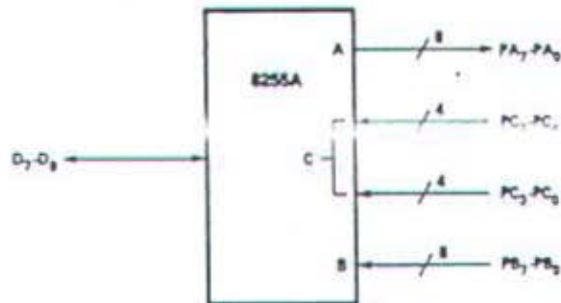
| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |



Programmable Peripheral Interface 8255

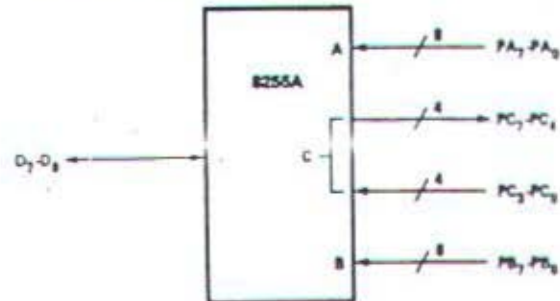
CONTROL WORD #7

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |



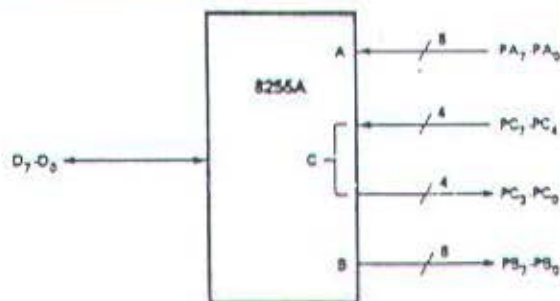
CONTROL WORD #11

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |



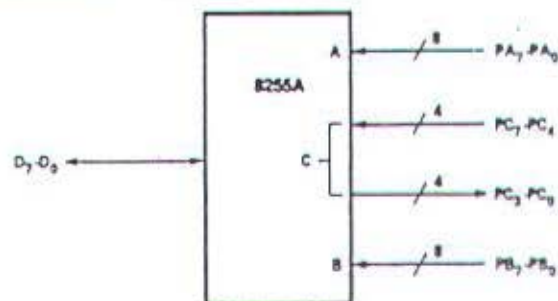
CONTROL WORD #12

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |



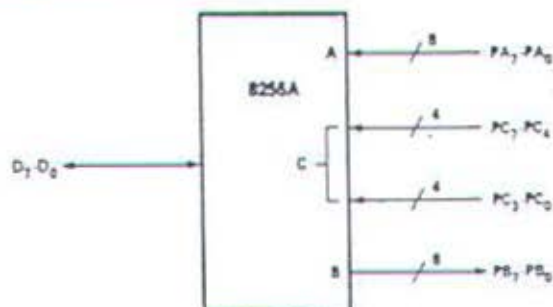
CONTROL WORD #14

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |



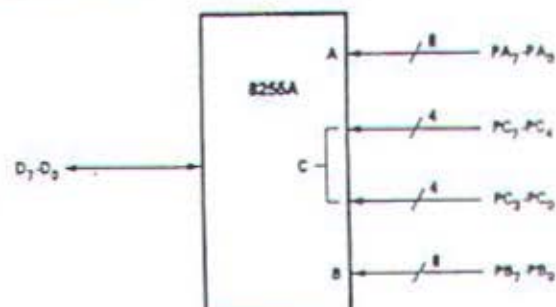
CONTROL WORD #13

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |



CONTROL WORD #15

| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |



Operating Modes

MODE 1 (Strobed Input/Output). This functional configuration provides a means for transferring I/O data to or from a specified port in conjunction with strobes or "handshaking" signals. In mode 1, port A and port B use the lines on port C to generate or accept these "handshaking" signals.

Mode 1 Basic Functional Definitions:

- Two Groups (Group A and Group B)
- Each group contains one 8-bit data port and one

- The 8-bit data port can be either input or output. Both inputs and outputs are latched.
- The 4-bit port is used for control and status of the 8-bit data port.

Input Control Signal Definition

STB (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F)

A "high" on this output indicates that the data has

8255 Programmable Peripheral Interface

been loaded into the input latch; in essence, an acknowledgement. IBF is set by \overline{STB} input being low and is reset by the rising edge of the \overline{RD} input.

INTR (Interrupt Request)

A "high" on this output can be used to interrupt the CPU when an input device is requesting service. INTR is set by the \overline{STB} input being low, IBF is a "one" and INTE is a "one". It is reset by the falling edge of \overline{RD} . This procedure allows an input device to request service from the CPU by simply strobing its data into the port.

INTE A

Controlled by bit set/reset of PC_4 .

INTE B

Controlled by bit set/reset of PC_2 .

Output Control Signal Definition

\overline{OBF} (Output Buffer Full F/F). The \overline{OBF} output will go "low" to indicate that the CPU has written data out to the specified port. The \overline{OBF} F/F will be set to the rising edge of the \overline{WR} input and reset by \overline{ACK} input being low.

\overline{ACK} (Acknowledge Input). A "low" on this input informs the 8255A that the data from port A or port B has been accepted. In essence, a response from the peripheral device indicating that it has received the data output by the CPU.

INTR (Interrupt Request). A "high" on this output can be used to interrupt the CPU when an output device has accepted data transmitted by the CPU. INTR is set when \overline{ACK} is a "one", \overline{OBF} is a "one" and INTE is a "one". It is reset by the falling edge of \overline{WR} .

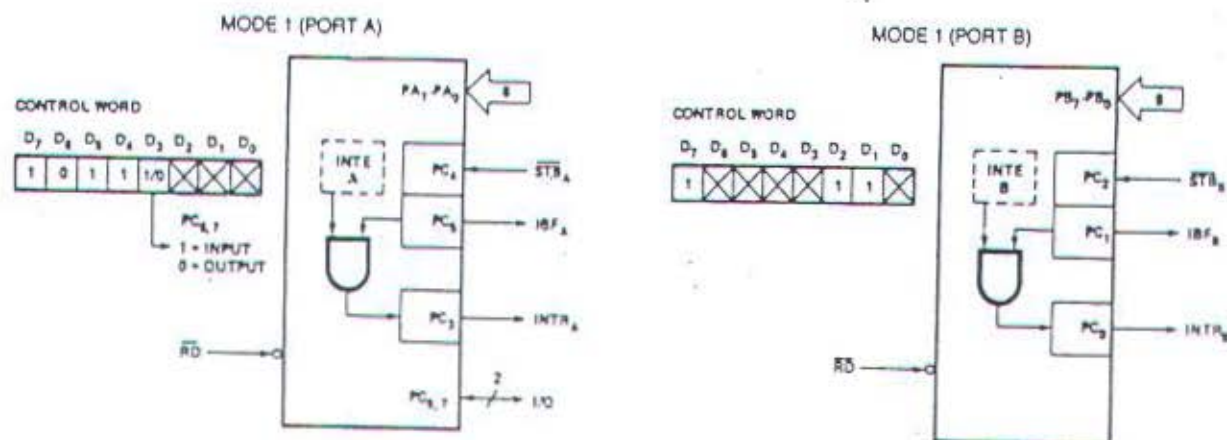
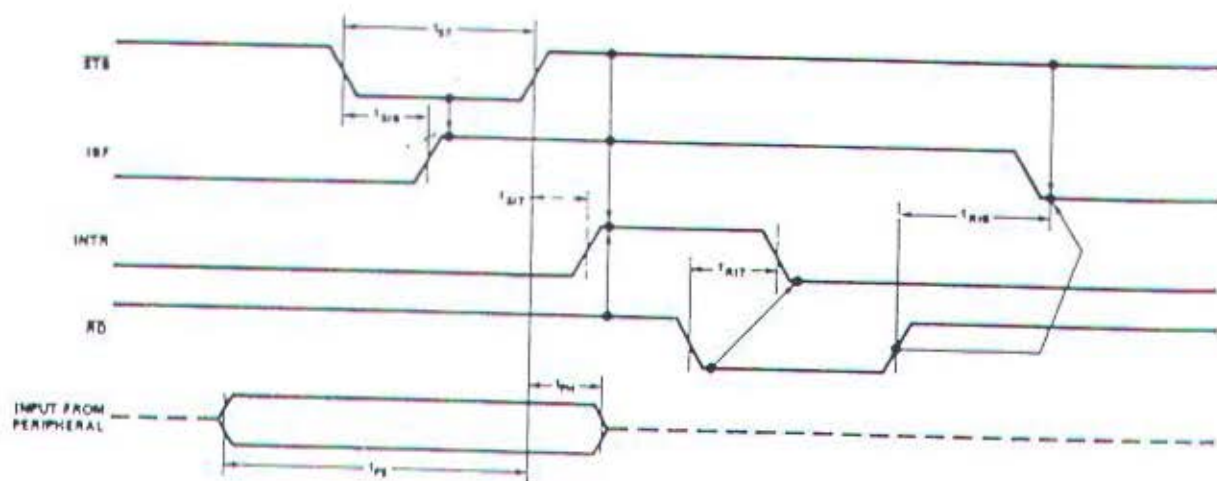


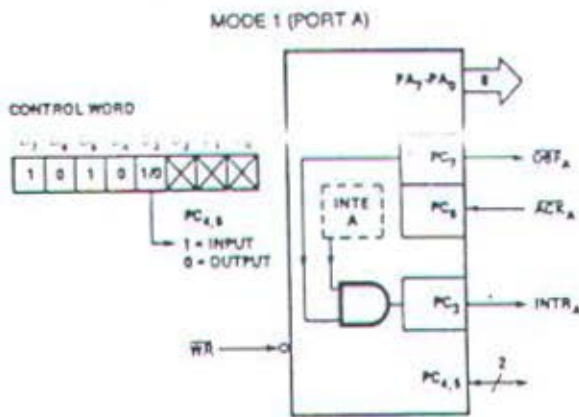
Figure 8. MODE 1 Input



Programmable Peripheral Interface 8255

INTE A

Controlled by bit set/reset of PC₅.



INTE B

Controlled by bit set/reset of PC₂.

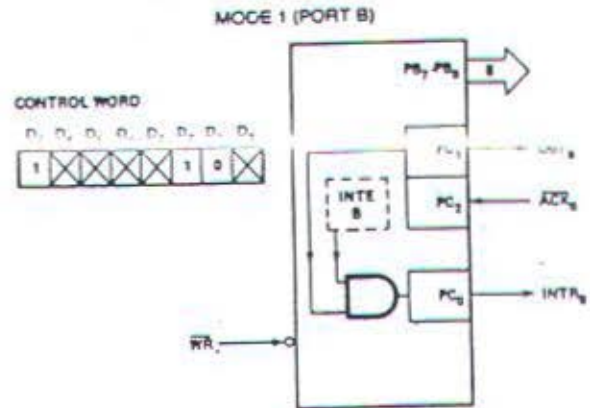


Figure 10. MODE 1 Output

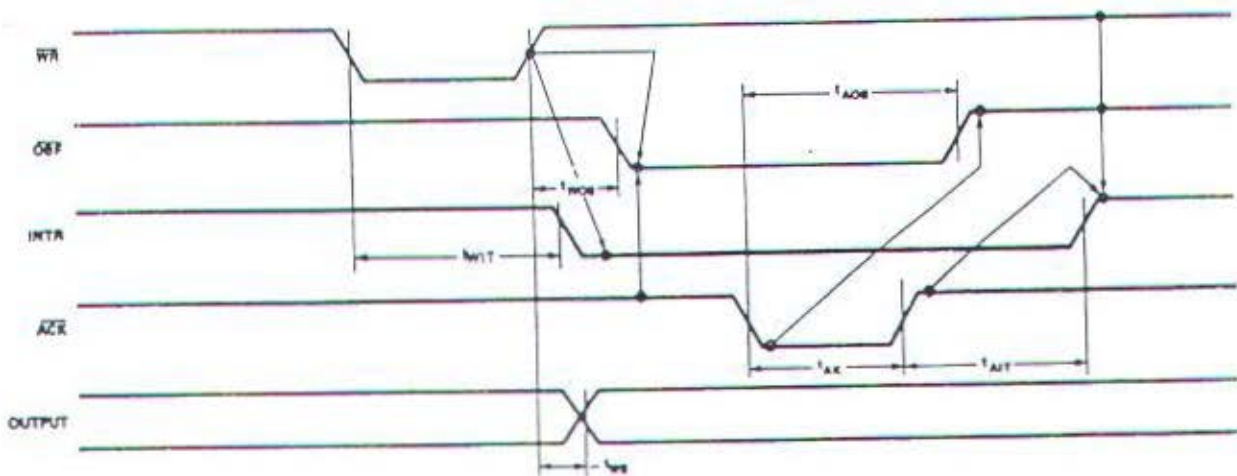
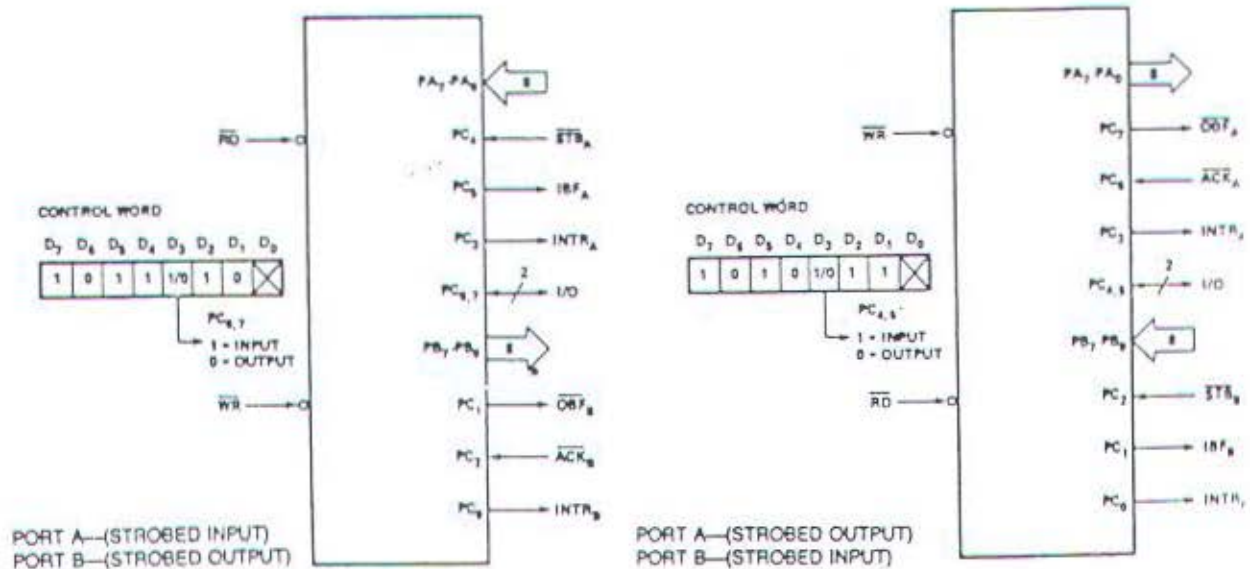


Figure 11. MODE 1 (Strobed Output)



8255 Programmable Peripheral Interface

Combinations of MODE 1

Port A and Port B can be individually defined as input or output in MODE 1 to support a wide variety of strobed I/O applications.

Operating Modes

MODE 2 (Strobed Bidirectional Bus I/O). This functional configuration provides a means for communicating with a peripheral device or structure on a single 8-bit bus for both transmitting and receiving data (bidirectional bus I/O). "Handshaking" signals are provided to maintain proper bus flow discipline in a similar manner to MODE 1. Interrupt generation and enable/disable functions are also available.

MODE 2 Basic Functional Definitions:

- Used in Group A only.
- One 8-bit, bi-directional bus Port (Port A) and a 5-bit control Port (Port C).
- Both inputs and outputs are latched.
- The 5-bit control port (Port C) is used for control and status for the 8-bit, bi-directional bus port (Port A).

Bidirectional Bus I/O Control Signal Definition

INTR (Interrupt Request). A high on this output can be used to interrupt the CPU for both input or output operations.

Output Operations

\overline{OBF} (Output Buffer Full). The \overline{OBF} output will go "low" to indicate that the CPU has written data out to port A.

\overline{ACK} (Acknowledge). A "low" on this input enables the tri-state output buffer of port A to send out the data. Otherwise, the output buffer will be in the high impedance state.

INTE 1 (The INTE Flip-Flop Associated with OBF). Controlled by bit set/reset of PC_6 .

Input Operations

\overline{STB} (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F). A "high" on this output indicates that data has been loaded into the input latch.

INTE 2 (The INTE Flip-Flop Associated with IBF).

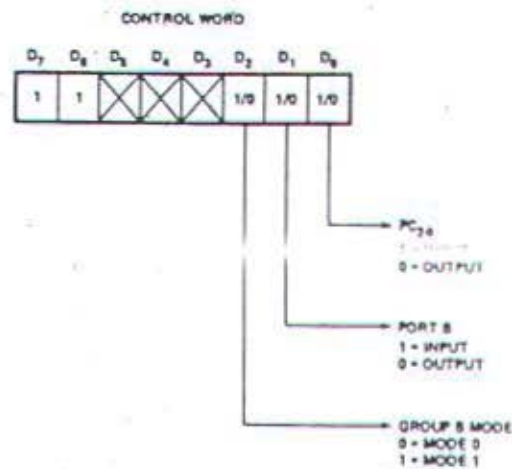


Figure 13. MODE Control Word

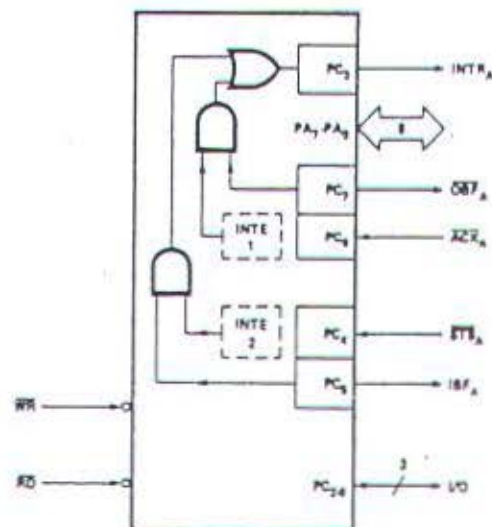
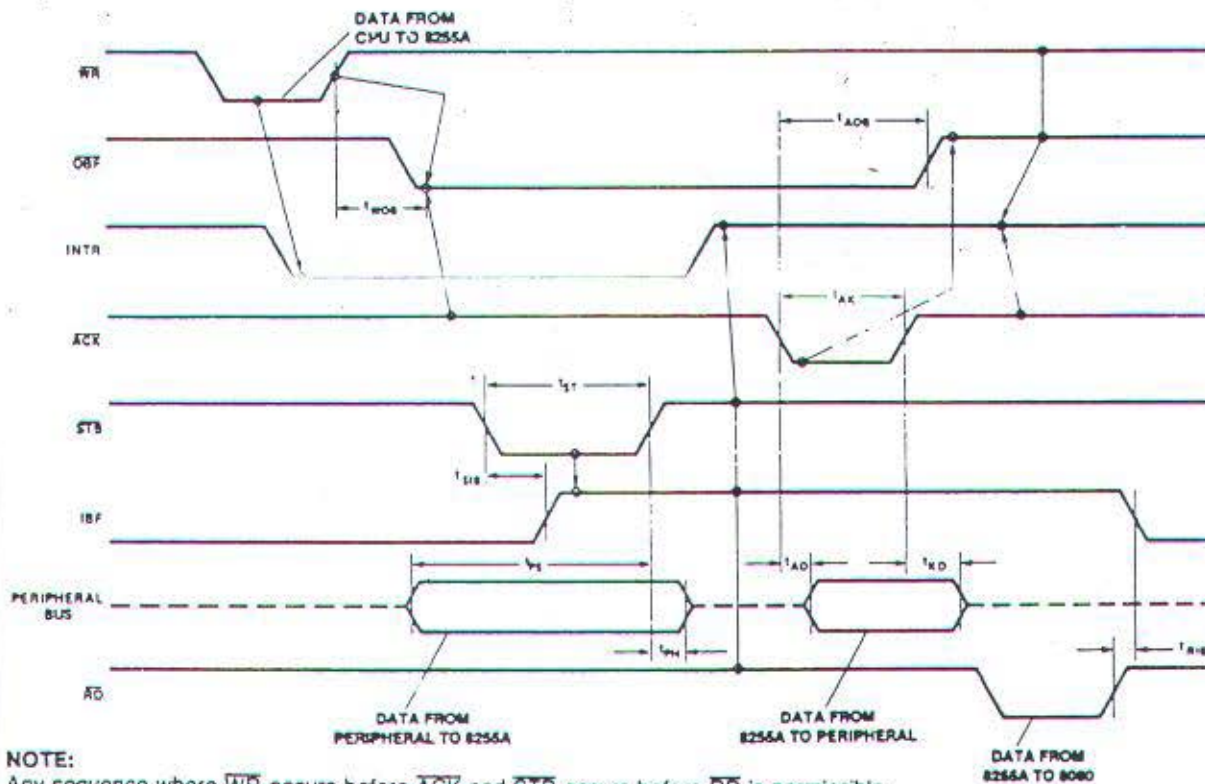


Figure 14. MODE 2

Programmable Peripheral Interface 8255



NOTE:

Any sequence where \overline{WR} occurs before \overline{ACK} and \overline{STB} occurs before \overline{RD} is permissible.
 $(\overline{INTR} = \overline{IBF} \cdot \overline{MASK} \cdot \overline{STB} \cdot \overline{RD} + \overline{OBF} \cdot \overline{MASK} \cdot \overline{ACK} \cdot \overline{WR})$

Figure 15. MODE 2 (Bidirectional)

Mode Definition Summary

| | MODE 0 | | MODE 1 | | MODE 2 |
|-----------------|--------|-----|-------------------|-------------------|-------------------|
| | IN | OUT | IN | OUT | GROUP A ONLY |
| PA ₀ | IN | OUT | IN | OUT | ↔ |
| PA ₁ | IN | OUT | IN | OUT | ↔ |
| PA ₂ | IN | OUT | IN | OUT | ↔ |
| PA ₃ | IN | OUT | IN | OUT | ↔ |
| PA ₄ | IN | OUT | IN | OUT | ↔ |
| PA ₅ | IN | OUT | IN | OUT | ↔ |
| PA ₆ | IN | OUT | IN | OUT | ↔ |
| PA ₇ | IN | OUT | IN | OUT | ↔ |
| PB ₀ | IN | OUT | IN | OUT | — |
| PB ₁ | IN | OUT | IN | OUT | — |
| PB ₂ | IN | OUT | IN | OUT | — |
| PB ₃ | IN | OUT | IN | OUT | — |
| PB ₄ | IN | OUT | IN | OUT | — |
| PB ₅ | IN | OUT | IN | OUT | — |
| PB ₆ | IN | OUT | IN | OUT | — |
| PB ₇ | IN | OUT | IN | OUT | — |
| PC ₀ | IN | OUT | INTR _B | INTR _B | I/O |
| PC ₁ | IN | OUT | IBF _B | IBF _B | I/O |
| PC ₂ | IN | OUT | STB _B | STB _B | I/O |
| PC ₃ | IN | OUT | INTR _A | INTR _A | INTR _A |
| PC ₄ | IN | OUT | STB _A | I/O | STB _A |
| PC ₅ | IN | OUT | IBF _A | I/O | IBF _A |
| PC ₆ | IN | OUT | I/O | ACK _A | ACK _A |
| PC ₇ | IN | OUT | I/O | OBF _A | OBF _A |

MODE 0 OR MODE 1 ONLY

8255 Programmable Peripheral Interface

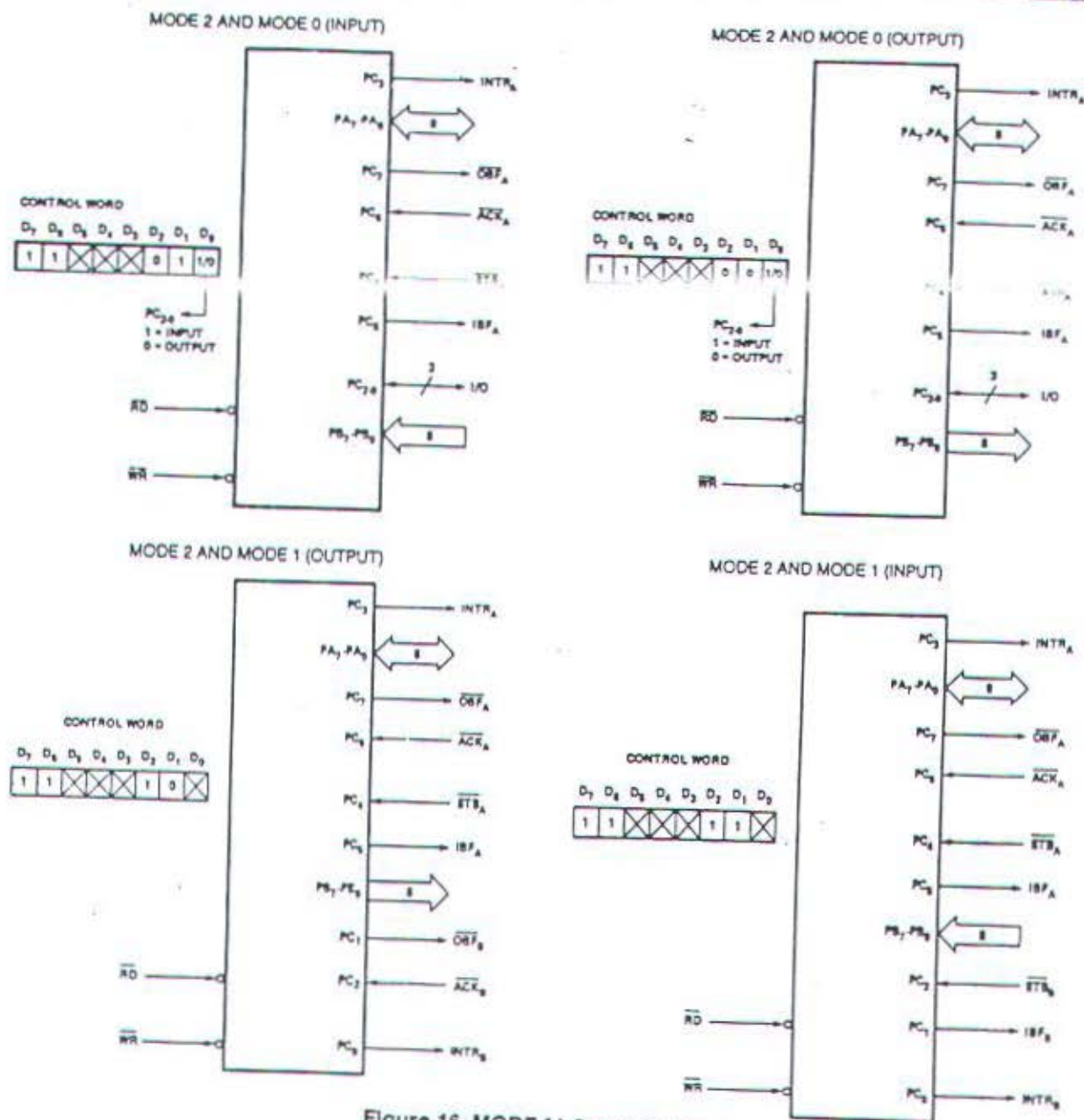


Figure 16. MODE 1/4 Combinations

Special Mode Combination Considerations

There are several combinations of modes when not all of the bits in Port C are used for control or status. The remaining bits can be used as follows:

If Programmed as Inputs—

All input lines can be accessed during a normal Port C read.

If Programmed as Outputs—

Bits in C upper (PC₇–PC₄) must be individually accessed using the bit set/reset function.

Bits in C lower (PC₃–PC₀) can be accessed using

the bit set/reset function or accessed as a three-some by writing into Port C.

Source Current Capability on Port B and Port C

Any set of eight output buffers, selected randomly from Ports B and C can source 1 mA at 1.5 volts.

This feature allows the 8255 to directly drive Darlington type drivers and high-voltage displays that require such source current.

Reading Port C Status

In Mode 0, Port C transfers data to or from the pe-

Programmable Peripheral Interface 8255

ipheral device. When the 8255 is programmed to function in Modes 1 or 2, Port C generates or accepts "hand-shaking" signals with the peripheral device. Reading the contents of Port C allows the programmer to test or verify the "status" of each peripheral device and change the program flow accordingly.

There is no special instruction to read the status information from Port C. A normal read operation of Port C is executed to perform this function.

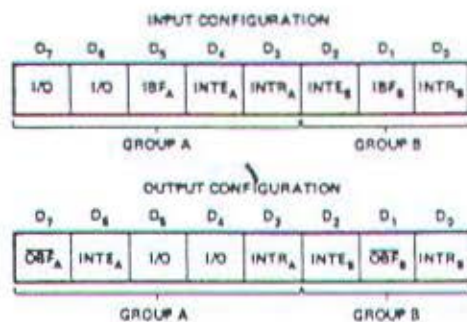


Figure 17. MODE 1 Status Word Format

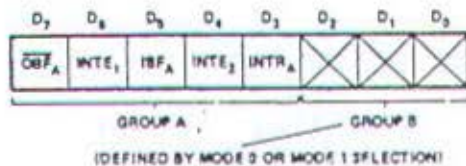


Figure 18. MODE 2 Status Word Format

APPLICATIONS OF THE 8255A

The 8255A is a very powerful tool for interfacing peripheral equipment to the microcomputer system. It represents the optimum use of available pins and is flexible enough to interface almost any I/O device without the need for additional external logic.

Each peripheral device in a microcomputer system usually has a "service routine" associated with it. The routine manages the software interface between the device and the CPU. The functional definition of the 8255A is programmed by the I/O service routine and becomes an extension of the system software. By examining the I/O devices interface characteristics for both data transfer and timing, and matching this information to the examples and tables in the detailed operational description, a control word can easily be developed to initialize the 8255A to exactly "fit" the application. Figures 19 through 25 represent a few examples of typical applications

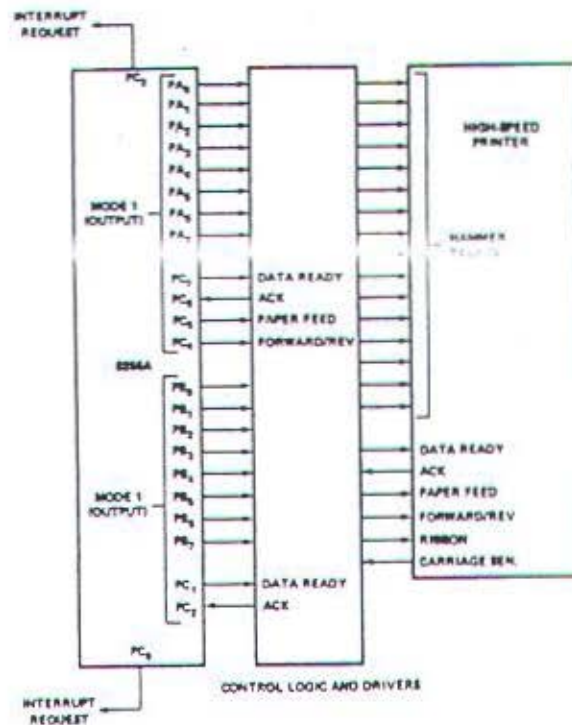
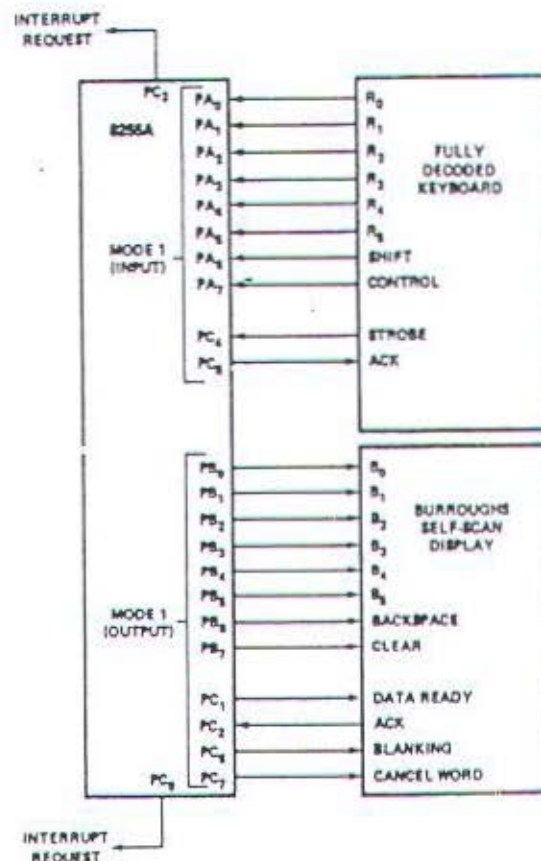


Figure 19. Printer Interface



8255 Programmable Peripheral Interface

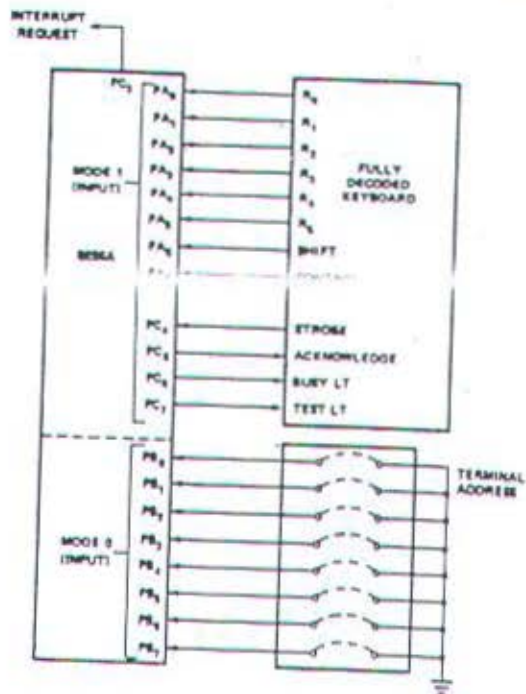


Figure 21. Keyboard and Terminal Address Interface

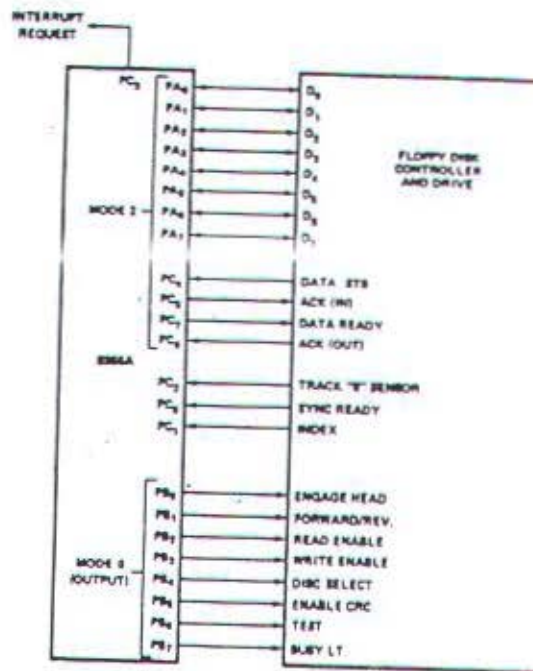


Figure 23. Basic Floppy Disk Interface

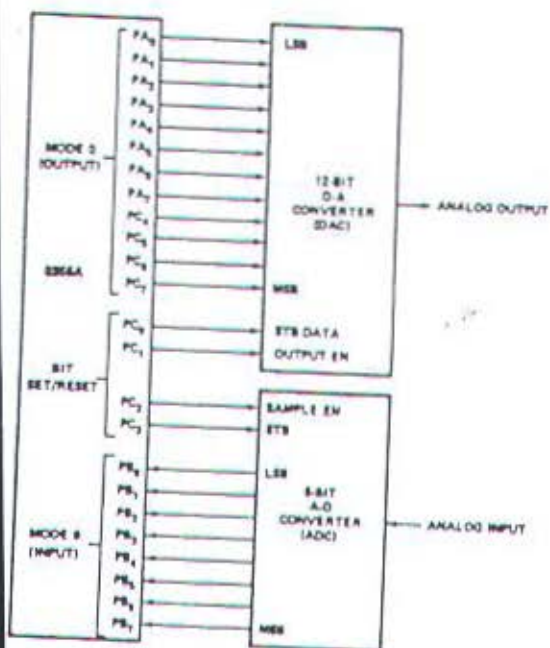
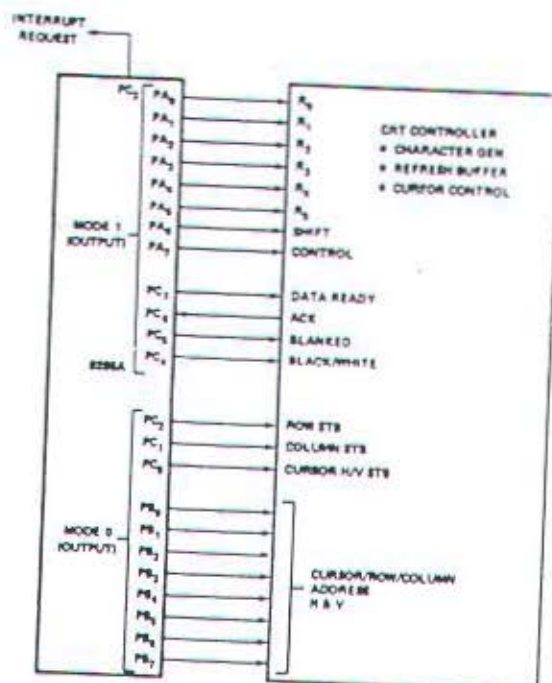


Figure 22. Digital to Analog, Analogs to Digital



Programmable Peripheral Interface 8255

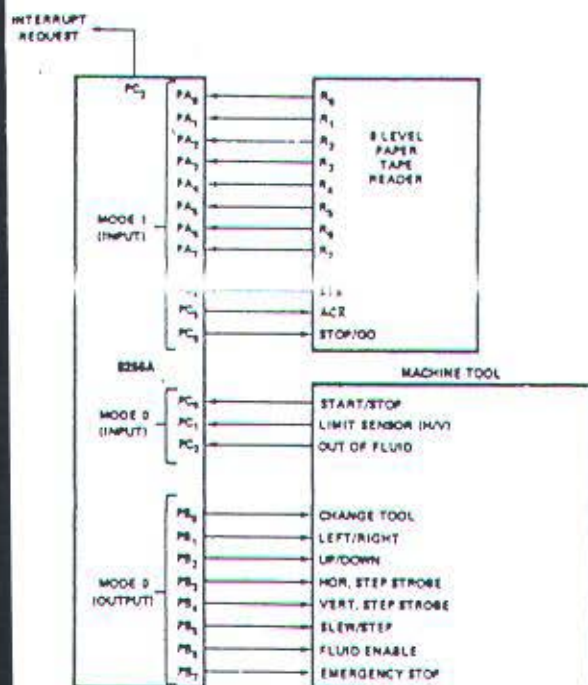


Figure 25. Machine Tool Controller Interface

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias 0°C to 70°C
 Storage Temperature -65°C to +150°C
 Voltage on Any Pin
 with Respect to Ground -0.5V to +7V
 Power Dissipation 1 Watt

**Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

D.C. CHARACTERISTICS $T_A = 0^\circ\text{C to } 70^\circ\text{C}$, $V_{CC} = +5\text{V} \pm 10\%$, $\text{GND} = 0\text{V}$ *

| Symbol | Parameter | Min | Max | Unit | Test Conditions |
|----------------------|---------------------------------------|------|----------|---------------|---|
| V_{IL} | Input Low Voltage | -0.5 | 0.8 | V | |
| V_{IH} | Input High Voltage | 2.0 | V_{CC} | V | |
| $V_{OL}(\text{DB})$ | Output Low Voltage (Data Bus) | | 0.45* | V | $I_{OL} = 2.5 \text{ mA}$ |
| $V_{OL}(\text{PER})$ | Output Low Voltage (Peripheral Port) | | 0.45* | V | $I_{OL} = 1.7 \text{ mA}$ |
| $V_{OH}(\text{DB})$ | Output High Voltage (Data Bus) | 2.4 | | V | $I_{OH} = -400 \mu\text{A}$ |
| $V_{OH}(\text{PER})$ | Output High Voltage (Peripheral Port) | 2.4 | | V | $I_{OH} = -200 \mu\text{A}$ |
| $I_{DAR}^{(1)}$ | Darlington Drive Current | -1.0 | -4.0 | mA | $R_{EXT} = 750\Omega$; $V_{EXT} = 1.5\text{V}$ |
| I_{CC} | Power Supply Current | | 120 | mA | |
| I_{IL} | Input Load Current | | ± 10 | μA | $V_{IN} = V_{CC} \text{ to } 0\text{V}$ |
| I_{OFL} | Output Float Leakage | | ± 10 | μA | $V_{OUT} = V_{CC} \text{ to } 0.45\text{V}$ |

NOTE:

1. Available on any 8 pins from Port B and C.

CAPACITANCE $T_A = 25^\circ\text{C}$, $V_{CC} = \text{GND} = 0\text{V}$

| Symbol | Parameter | Min | Typ | Max | Unit | Test Conditions |
|-----------|-------------------|-----|-----|-----|------|--|
| C_{IN} | Input Capacitance | | | 10 | pF | $f_c = 1 \text{ MHz}^{(4)}$ |
| $C_{I/O}$ | I/O Capacitance | | | 20 | pF | Unmeasured pins returned to GND ⁽⁴⁾ |

8255 Programmable Peripheral Interface

A.C. CHARACTERISTICS $T_A = 0^\circ\text{C to } 70^\circ\text{C}$, $V_{CC} = +5\text{V} \pm 10\%$, $\text{GND} = 0\text{V}^*$

Bus Parameters

READ

| Symbol | Parameter | 8255A | | 8255A-5 | | Unit |
|----------|-------------------------------------|-------|-----|---------|-----|------|
| | | Min | Max | Min | Max | |
| t_{AH} | Address Stable before READ | 0 | | 0 | | ns |
| t_{RA} | Address Stable after READ | 0 | | 0 | | ns |
| t_{RR} | READ Pulse Width | 300 | | 300 | | ns |
| t_{RD} | Data Valid from READ ⁽¹⁾ | | 250 | | 200 | ns |
| t_{DF} | Data Float after READ | 10 | 150 | 10 | 100 | ns |
| t_{RV} | Time between READs and/or WRITEs | 850 | | 850 | | ns |

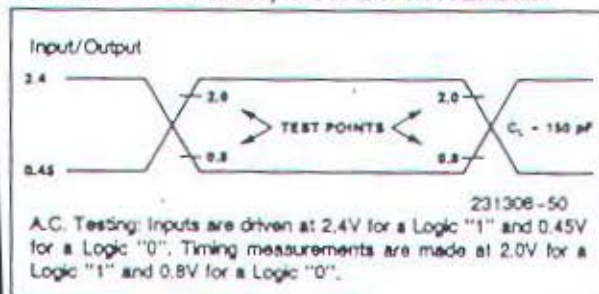
WRITE

| Symbol | Parameter | 8255A | | 8255A-5 | | Unit |
|----------|-----------------------------|-------|-----|---------|-----|------|
| | | Min | Max | Min | Max | |
| t_{AW} | Address Stable before WRITE | 0 | | 0 | | ns |
| t_{WA} | Address Stable after WRITE | 20 | | 20 | | ns |
| t_{WW} | WRITE Pulse Width | 400 | | 300 | | ns |
| t_{DW} | Data Valid to WRITE (T.E.) | 100 | | 100 | | ns |
| t_{WD} | Data Valid after WRITE | 30 | | 30 | | ns |

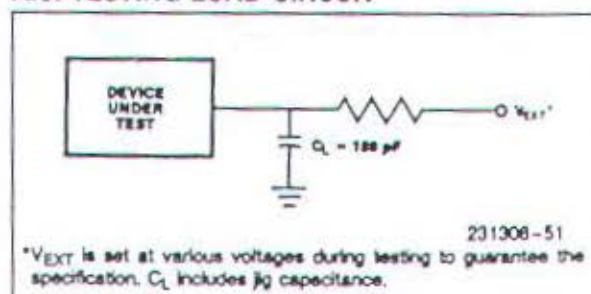
OTHER TIMINGS

| Symbol | Parameter | 8255A | | 8255A-5 | | Unit |
|----------|----------------------------------|-------|-----|---------|-----|------|
| | | Min | Max | Min | Max | |
| t_{WB} | WR = 1 to Output ⁽¹⁾ | | 350 | | 350 | ns |
| t_{IR} | Peripheral Data before RD | 0 | | 0 | | ns |
| t_{HR} | Peripheral Data after RD | 0 | | 0 | | ns |
| t_{AK} | ACK Pulse Width | 300 | | 300 | | ns |
| t_{ST} | STB Pulse Width | 500 | | 500 | | ns |
| t_{PS} | Per. Data before T.E. of STB | 0 | | 0 | | ns |
| t_{PH} | Per. Data after T.E. of STB | 180 | | 180 | | ns |
| t_{AD} | ACK = 0 to Output ⁽¹⁾ | | 300 | | 300 | ns |
| t_{KD} | ACK = 1 to Output Float | 20 | 250 | 20 | 250 | ns |

A.C. TESTING INPUT, OUTPUT WAVEFORM



A.C. TESTING LOAD CIRCUIT



A.C. CHARACTERISTICS (Continued)

OTHER TIMINGS (Continued)

| Symbol | Parameter | 8255A | | 8255A-5 | | Unit |
|-----------|--------------------------|-------|-----|---------|-----|------|
| | | Min | Max | Min | Max | |
| t_{WOB} | WR = 1 to OBF = 0(1) | | 650 | | 650 | ns |
| t_{AOB} | ACK = 0 to OBF = 1(1) | | 350 | | 350 | ns |
| t_{SIB} | STB = 0 to IBF = 1(1) | | 300 | | 300 | ns |
| t_{RIB} | RD = 1 to IBF = 0(1) | | 300 | | 300 | ns |
| t_{RIT} | RD = 0 to INTR = 0(1) | | 400 | | 400 | ns |
| t_{SIT} | STB = 1 to INTR = 1(1) | | 300 | | 300 | ns |
| t_{AIT} | ACK = 1 to INTR = 1(1) | | 350 | | 350 | ns |
| t_{WIT} | WR = 0 to INTR = 0(1, 3) | | 850 | | 850 | ns |

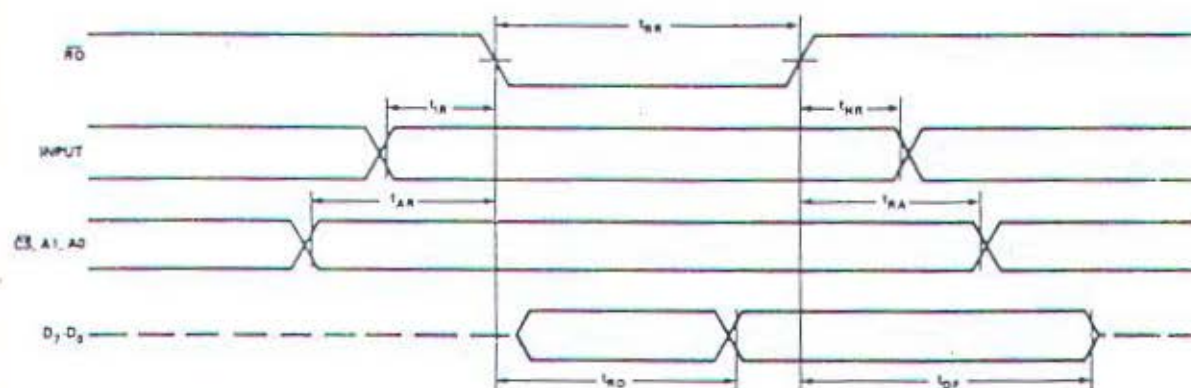
NOTES:

1. Test Conditions: $C_L = 150$ pF.
2. Period of Reset pulse must be at least 50 μ s during or after power on. Subsequent Reset pulse can be 500 ns min.
3. INTR \uparrow may occur as early as WR \downarrow .
4. Sampled, not 100% tested.

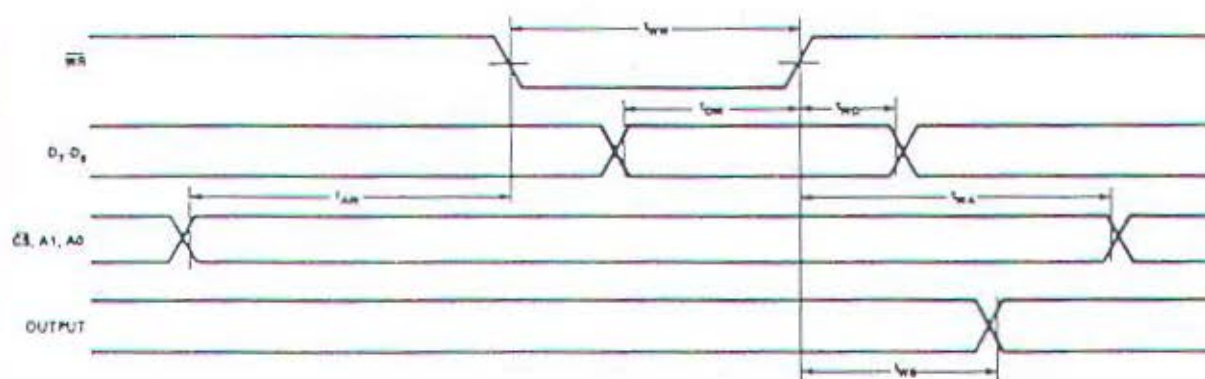
*For Extended Temperature EXPRESS, use M8255A electrical parameters.

WAVEFORMS

MODE 0 (BASIC INPUT)



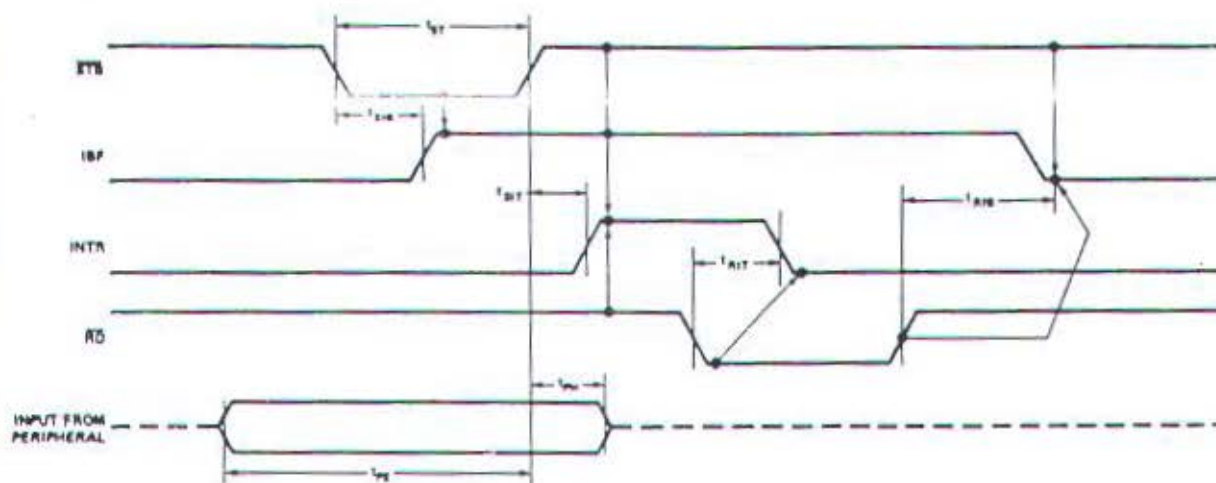
MODE 0 (BASIC OUTPUT)



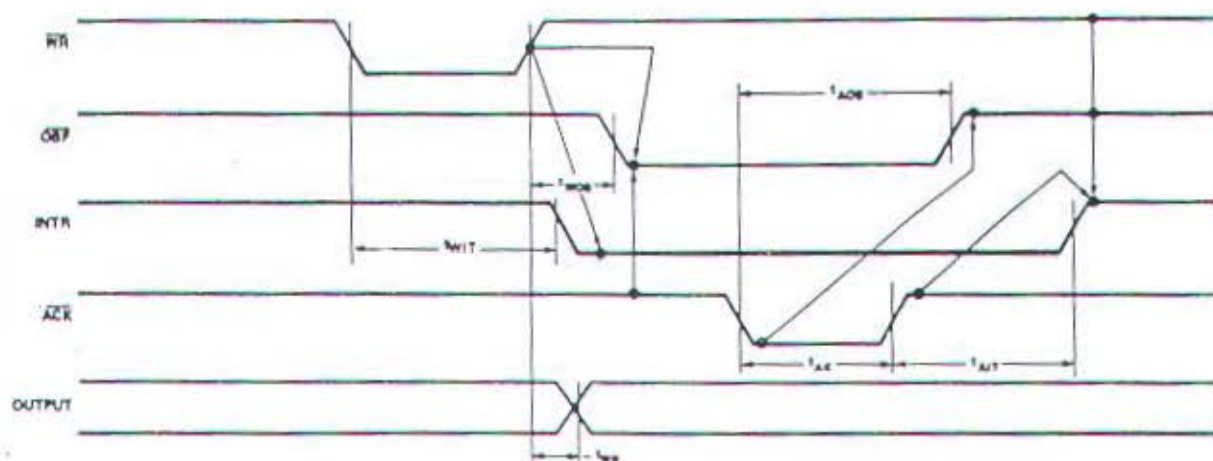
8255 Programmable Peripheral Interface

WAVEFORMS (Continued)

MODE 1 (STROBED INPUT)



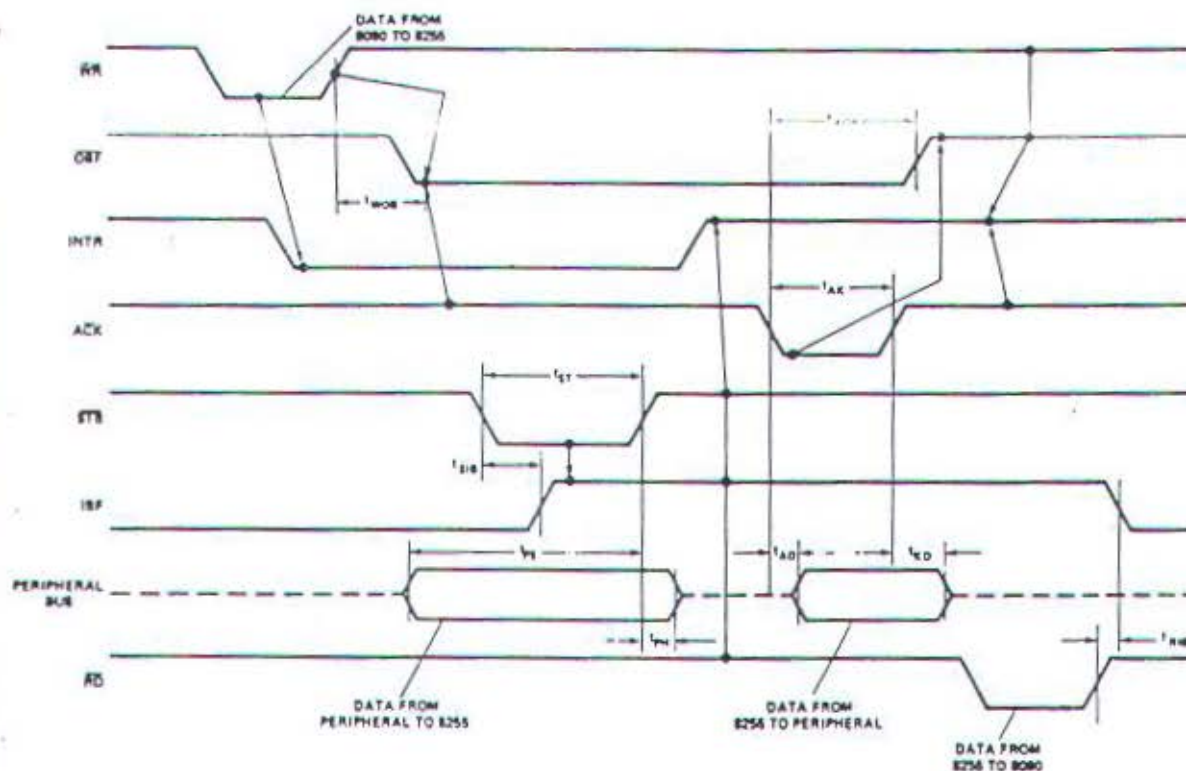
MODE 1 (STROBED OUTPUT)



Programmable Peripheral Interface 8255

WAVEFORMS (Continued)

MODE 2 (BIDIRECTIONAL)



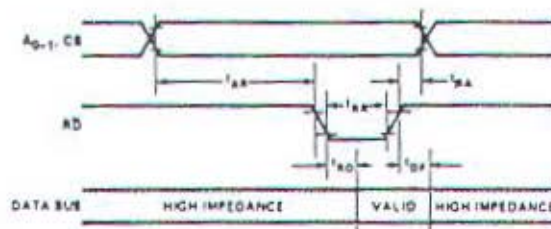
NOTE:

Any sequence where \overline{WR} occurs before \overline{ACK} and \overline{STB} occurs before \overline{RD} is permissible.
 (INTR = IBF • MASK • \overline{STB} • RD + \overline{OBT} • MASK • \overline{ACK} • \overline{WR})

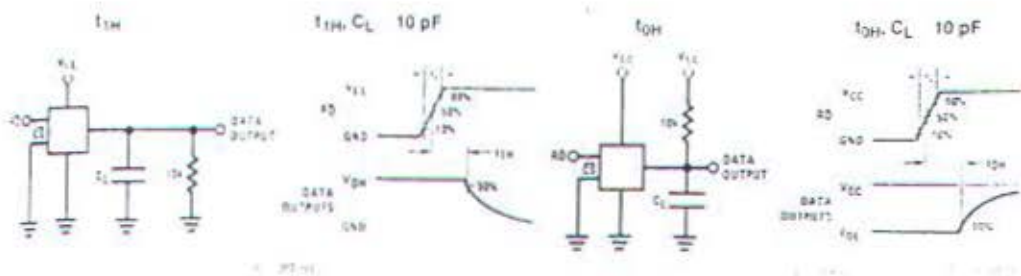
WRITE TIMING



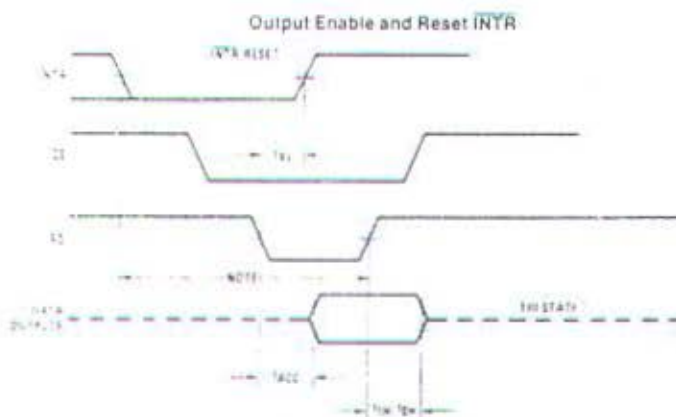
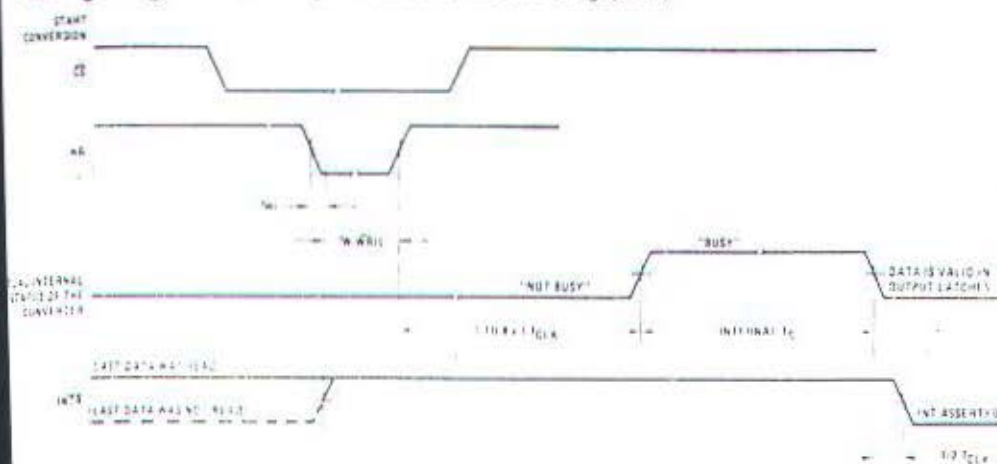
READ TIMING



TRI-STATE Test Circuits and Waveforms



Timing Diagrams (All timing is measured from the 50% voltage points)



When Reset occurs, the output block pointers will be 0 after assertion of interrupt to guarantee reset of INTR.

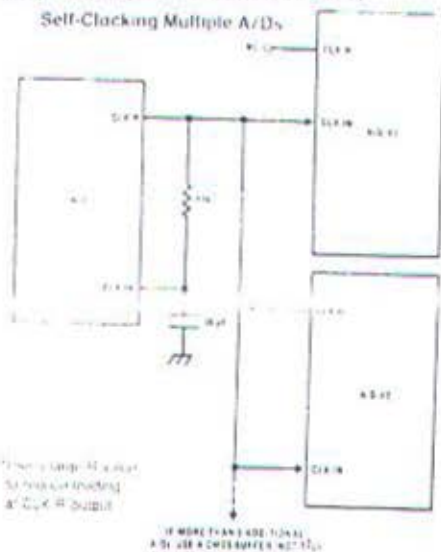
FIGURE 2

ADC0801/ADC0802/ADC0803/ADC0804/ADC0805

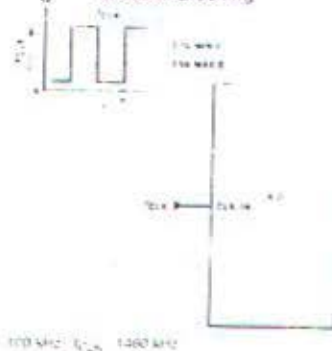
2

Typical Applications (Continued)

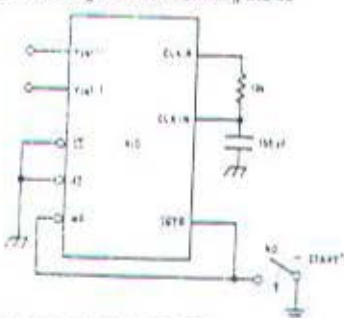
Self-Clocking Multiple A/Ds



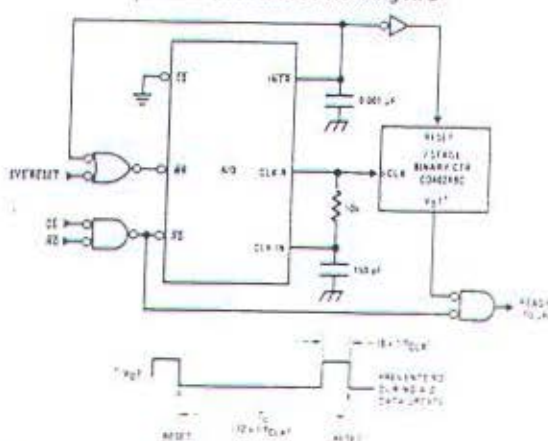
External Clocking



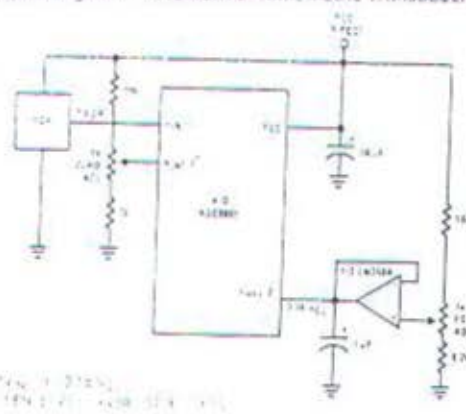
Self-Clocking in Free-Running Mode



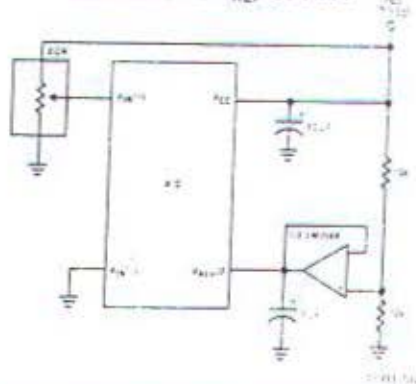
µP Interface for Free-Running A/D



Operating with "Automotive" Ratio-Metric Transducers

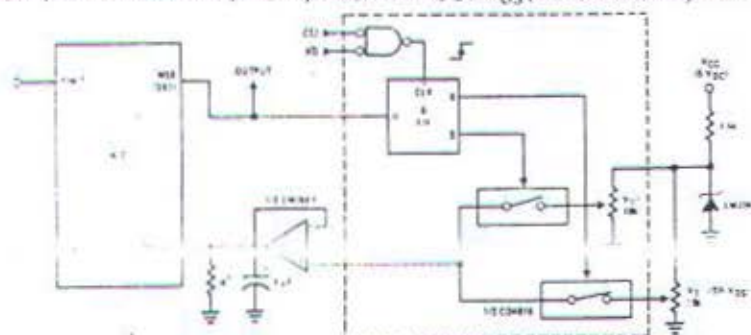


Ratio-Metric with $V_{REF}/2$ Forced



Typical Applications (Continued)

μ P-Compatible Differential-Input Comparator with Pre-Set V_{OS} (with or without Hysteresis)



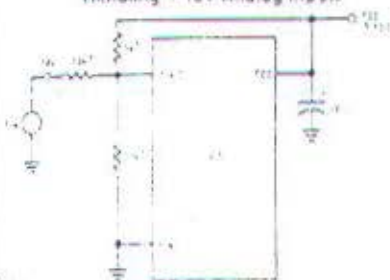
* See Figure 8-10 to select R value

$$OSR = \frac{1}{2} \ln \left(\frac{V_{OS}}{V_{OS} - V_{OSR}} \right)$$

OSR = Offset Voltage Ratio

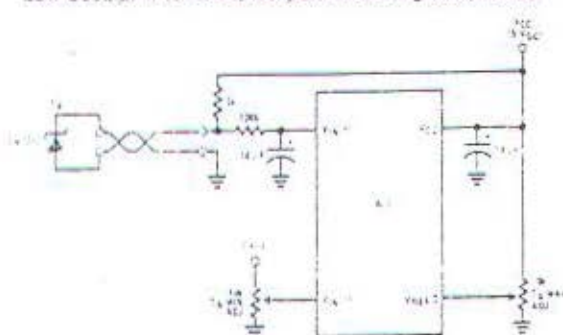
OSR = Offset Voltage Ratio

Handling $\pm 10V$ Analog Inputs

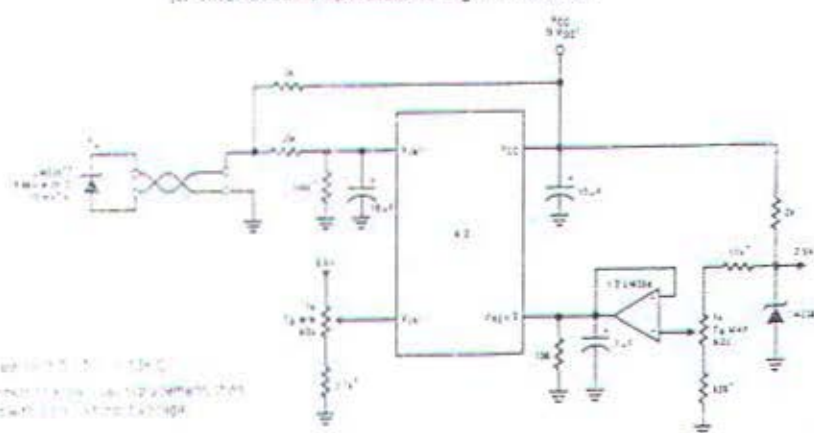


Resistor values are given in Figure 8-10.

Low-Cost, μ P Interfaced, Temperature-to-Digital Converter



μ P Interfaced Temperature-to-Digital Converter

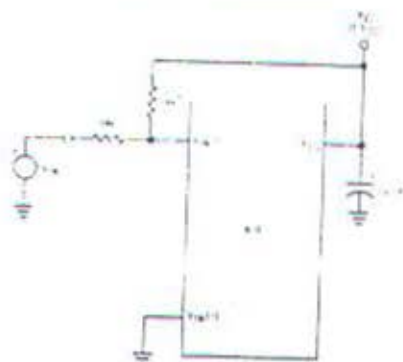


Resistor values are given in Figure 8-10.

Figure 8-10-E

Typical Applications (Continued)

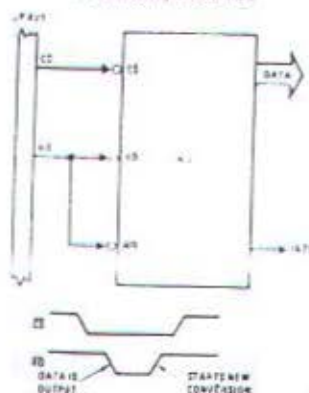
Handling 5V Analog Inputs



IC: 74181, 74180

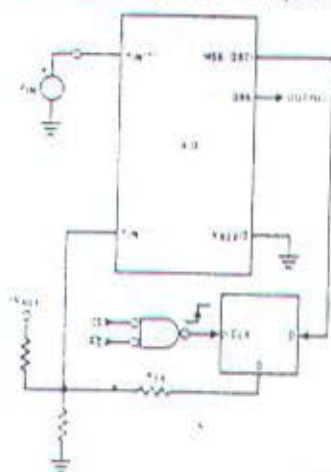
Source: *Practical Applications of the 74181 ALU*

Read-Only Interface



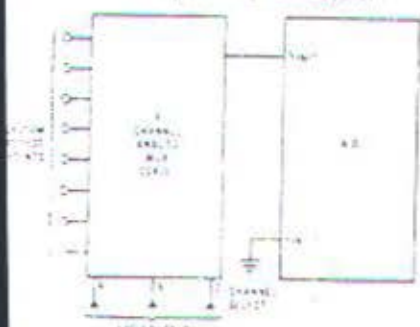
IC: 74181, 74180

μP Interfaced Comparator with Hysteresis



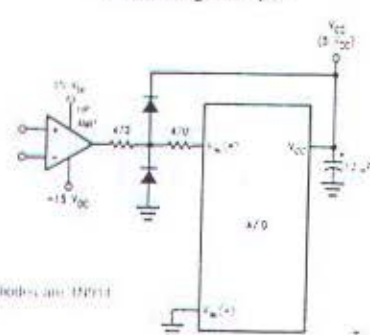
IC: 74181, 74180

Analog Self-Test for a System



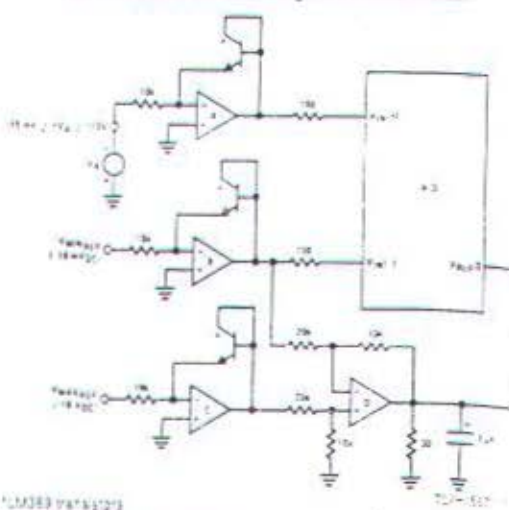
IC: 74181, 74180

Protecting the Input



Notes: 1. 74181

A Low-Cost, 3-Decade Logarithmic Converter



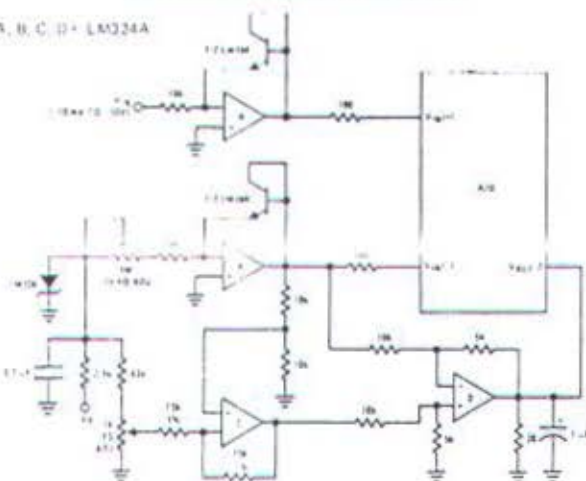
LM389 10k/10k/10k

A, B, C, D - LM324A quad op amp

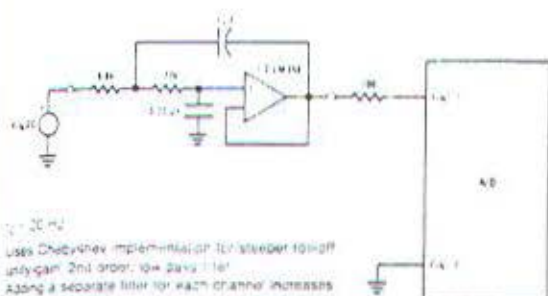
Typical Applications (Continued)

3-Decade Logarithmic A/D Converter

A, B, C, D = LM324A

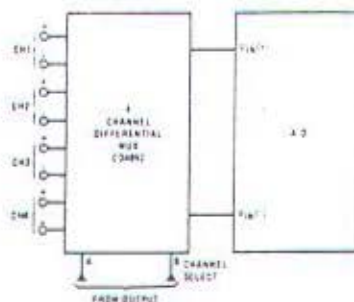


Noise Filtering the Analog Input

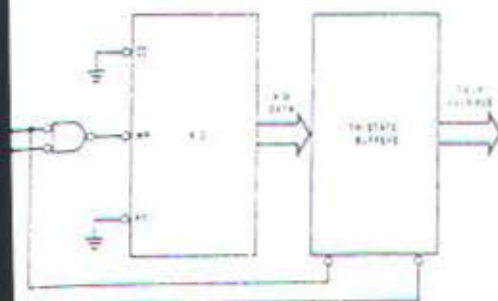


$f_c = 20 \text{ kHz}$
Uses Debye's implementation for slewer to off
only gain 2nd order, low pass filter
Adding a separate filter for each channel increases
system response time if an analog multiplexer
is used

Multiplexing Differential Inputs

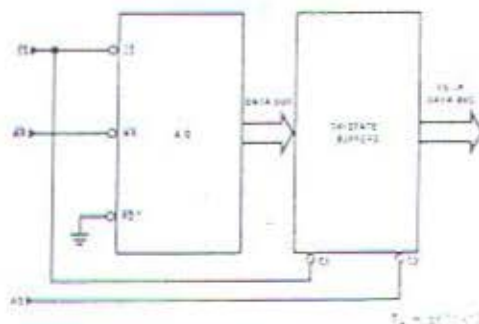


Output Buffers with A/D Data Enabled



16:1 D output data is updated 1 CLK period
prior to assertion of \overline{CS}

Increasing Bus Drive and/or Reducing Time on Bus



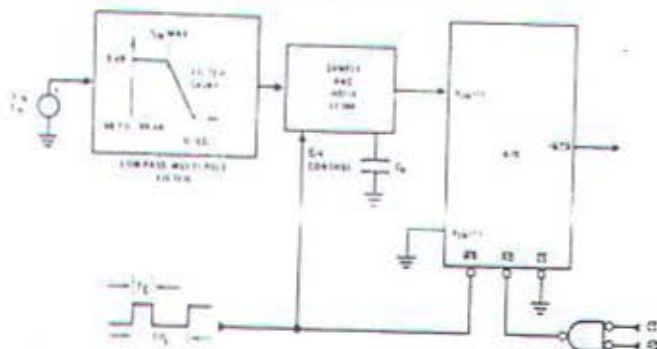
* Allows output data to set-up at falling edge of \overline{CS}

ADCC0801/ADCC0802/ADCC0803 ADCC0804/ADCC0805

2

Typical Applications µVermicrosoft

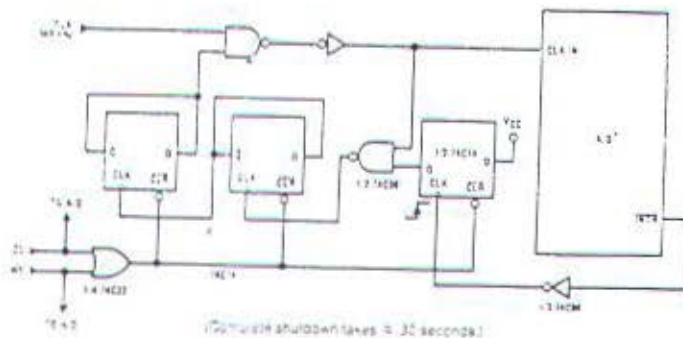
Sampling an AC Input Signal



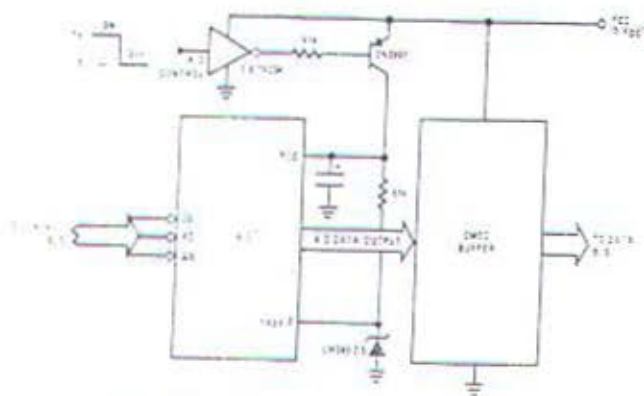
Note 1: Oversample whenever possible ($f_{\text{clk}} > 2(f_{\text{in}} + 60)$) to eliminate input frequency folding (aliasing) and to allow for the fast response of the filter.

Note 2: Consider the amplitude errors which are introduced within the passband of the filter.

70% Power Savings by Clock Gating



Power Savings by A/D and V_{REF} Shutdown



DOI: 10.1002/anie.200600011

Find I_{avg} and I_{rms} for a half-wave rectified $i(t) = I_m \sin \omega t$ with a DC supply at zero volts.

^a This program data was from a running output of A/D when in shutdown mode.

$$T_{\text{eff}} = m_{\text{eff}} \frac{d\mathbf{r}}{dt} + \mathbf{r} \frac{dm_{\text{eff}}}{dt}$$

Functional Description

UNDERSTANDING A/D ERROR SPECS

A perfect A/D transfer characteristic (staircase waveform) is shown in Figure 1a. The horizontal scale is analog input voltage and the particular points labeled are in steps of 1 LSB (19.53 mV with 2.5V tied to the V_{REF} (2 pins). The digital output codes, that correspond to these inputs, are shown as 0, 1, 2, and 3. For the perfect A/D, not only will center-valued inputs produce the correct output digital codes, but also each riser (the transitions between adjacent output codes) will be located $\pm 1/2$ LSB away from each center-value. As shown, the risers are ideal

for a range of analog input voltages that extend $\pm 1/2$ LSB from the ideal center-values. Each tread (the range of analog input voltage that provides the same digital output code) is therefore 1 LSB wide.

Figure 1b shows a worst case error plot for the ADC0801. Center-valued inputs are guaranteed to produce the correct output codes and the adjacent risers are guaranteed to be no closer to the center-value points than $\pm 1/4$ LSB. In

other words, if we apply an analog input equal to the center-value $\pm 1/4$ LSB, we guarantee that the A/D will produce the correct digital code. The maximum range of the position of the code transition is indicated by the horizontal arrow and it is guaranteed to be no more than $1/2$ LSB.

The error curves of Figure 1c shows a worst case error plot for the ALX8002. Here we guarantee that if we apply an analog input equal to the 1 LSB analog voltage center-value, the A/D will produce the correct digital code.

Next to each transfer function is shown the corresponding error plot. Many people may be more familiar with error plots than transfer functions. This plot shows the quantization error, which is provided by either a linear ramp or by the discrete output steps of a high resolution DAC. Notice that the error is continuously displayed and includes the quantization uncertainty of the A/D. For example the error at point 1 of Figure 1a is $+1/2$ LSB because the digital code appeared $1/2$ LSB in advance of the center-value of the tread. The error plots always have a constant negative slope and the abrupt up-side steps are always 1 LSB in magnitude.

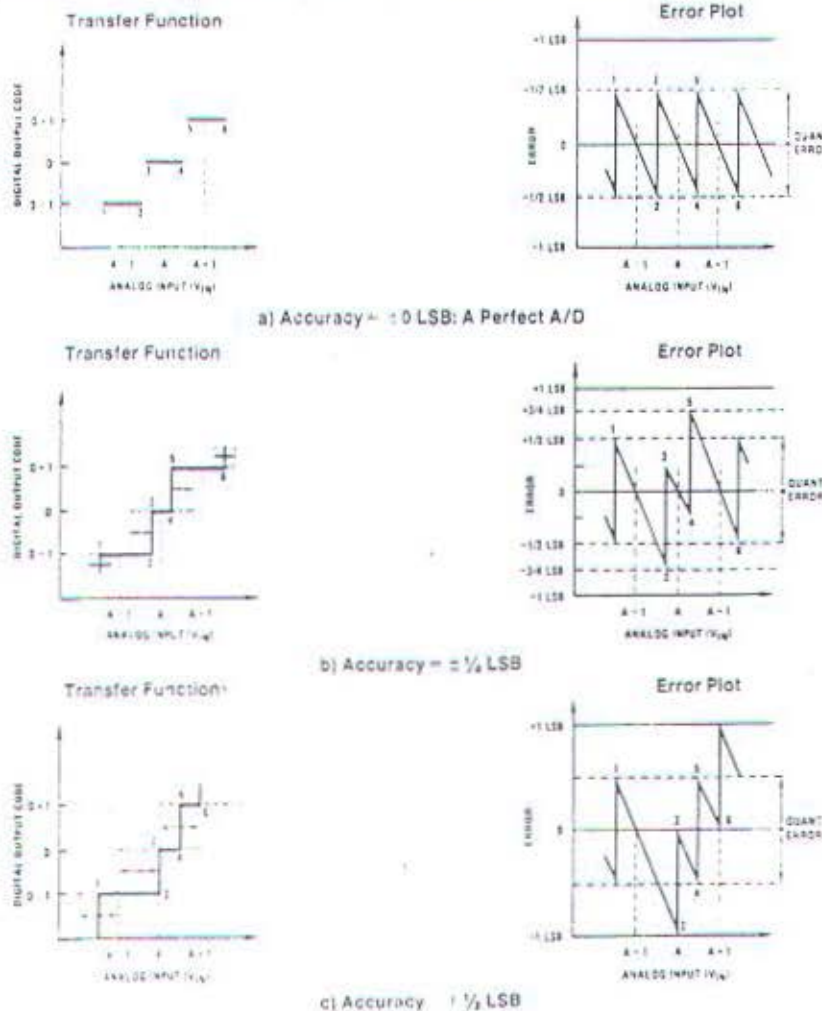


FIGURE 1. Clarifying the Error Specs of an A/D Converter

Functional Description (Continued)

2.0 FUNCTIONAL DESCRIPTION

The ADC0801 series contains a circuit equivalent of the 256R network. Analog switches are sequenced by successive approximation logic to match the analog difference input voltage $(V_{IN+} - V_{IN-})$ to a corresponding tap on the R network. The most significant bit is tested first and after 8 comparisons (64 clock cycles) a digital 8-bit binary code (1111 1111 = full-scale) is transferred to an output latch and then an interrupt is asserted (INTR makes a high-to-low transition). A conversion in process can be interrupted by issuing a second start command. This begins the

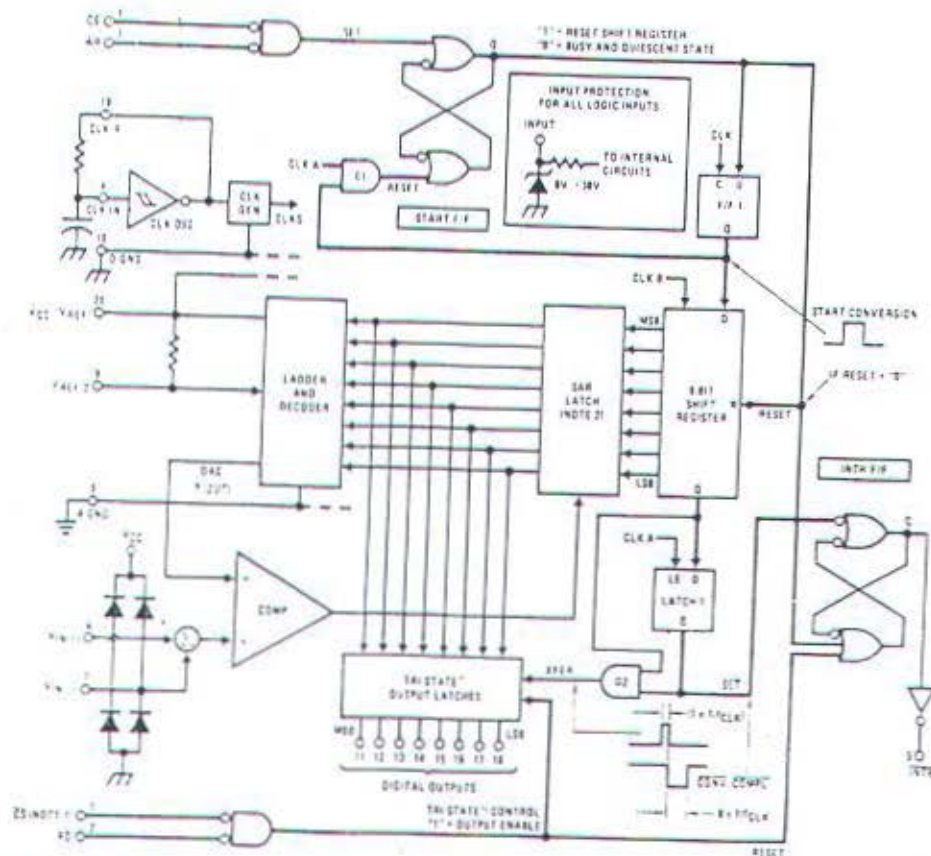
conversion by driving inputs by connecting INH to the \overline{WR} input with $\overline{CS} = 0$. To ensure start-up under all possible conditions, an external \overline{WR} pulse is required during the first power-up cycle.

On the high-to-low transition of the \overline{WR} input the internal SAR latches and the shift register stages are reset. As long as the \overline{CS} input and \overline{WR} input remain low, the A/D will remain in a reset state. Conversion will start from 1 to 8 clock periods after at least one of these inputs makes a low-to-high transition.

A functional diagram of the A/D converter is shown in Figure 2. All of the package pins are shown and the major logic control paths are drawn in heavier weight lines.

The converter is latched by having \overline{CS} and \overline{WR} simultaneously low. This sets the start flip-flop (F/F) and the resulting "1" level resets the 8-bit shift register, resets the interrupt (INTR) F/F and inputs a "1" to the D flip-flop, F/F1, which is at the input end of the 8-bit shift register. Internal clock signals then transfer this "1" to the Q output of F/F1. The AND gate, G1, combines this "1" output with a clock signal to provide a reset signal to the start F/F. If this "1" signal

is present, the start F/F is reset and the 8-bit shift register then can have the "1" clocked in, which starts the conversion process. If the signal were to still be present, this reset pulse would have no effect (both outputs of the start F/F would momentarily be at a "1" level) and the 8-bit shift register would continue to be held in the reset mode. This logic therefore allows for wide \overline{CS} and \overline{WR} signals and the converter will start after at least one of these signals returns high and the internal clocks again provide a reset signal for the start F/F.



Note 1: \overline{CS} shown twice for clarity.

Note 2: SAR = Successive Approximation Register.

FIGURE 2. Block Diagram

Functional Description (continued)

The "1" is clocked through the 8-bit shift register which completes the SAR search) it appears as the input to D-type latch, LATCH 1. As soon as this "1" is output from the shift register, the AND gate, G2, causes the now high signal to be fed back to the 11011111 output latch, LATCH 1. This subsequently enables the Q output as a high-to-low transition which causes the INTR F/F to set. An inverting buffer then supplies the INTR input signal.

When this SET control of the INTR F/F remains low for the external clock periods (as the internal clocks run at the frequency of the external clock). If the data output is continuously enabled (\overline{CS} and \overline{RD} both held low), the output will set at the beginning of each clock cycle.

transitions, because the SET input can control the Q output of the INTR F/F even though the RESET input is only at a "1" level in this operating mode. This INTR signal will therefore stay low for the duration of the SET which is 8 periods of the external clock frequency during the A/D is not started during this interval).

When operating in the free-running or continuous conversion mode (INTR pin tied to \overline{WR} and \overline{CS} wired low—see section 2.8), the START F/F is SET by the high-to-low transition of the INTR signal. This resets the SHIFT REGISTER, which causes the input to the D-type latch, LATCH 1, to go low. As the latch enable input is still present, the Q will go high, which then allows the INTR F/F to be set. This reduces the width of the resulting INTR output to only a few propagation delays (approximately 300

ps) as is to be read; the combination of both \overline{CS} and \overline{RD} going low will cause the INTR F/F to be reset and the A/D output latch will be enabled to provide the 8-bit output.

Digital Control Inputs

Digital control inputs (\overline{CS} , \overline{RD} , and \overline{WR}) meet standard 5V voltage levels. These signals have been renamed compared to the standard A/D Start and Output Enable. In addition, these inputs are active low to allow interfacing to microprocessor control buses. For microprocessor based applications, the \overline{CS} input (pin 1) is grounded and the standard A/D Start function is by an active low pulse applied at the \overline{WR} input (pin 7). The Output Enable function is caused by an active low pulse at the \overline{RD} input (pin 2).

Analog Differential Voltage Inputs and Common-Mode Rejection

The device has additional applications flexibility due to the differential voltage input. The $V_{IN(-)}$ input (pin 7) is used to automatically subtract a fixed voltage value from input readings (tare correction). This is also useful in current-to-voltage conversion. In addition, common-mode rejection can be reduced by use of the differential input mode between sampling $V_{IN(+)}$ and $V_{IN(-)}$ (pins 4 and 7). The maximum error voltage due to this

slight time difference between the input voltage samples is given by:

$$\Delta V_{IN(MAX)} = (V_P)(2\pi f_{CM}) \left(\frac{4.5}{f_{CLK}} \right)$$

where:

ΔV_{IN} is the error voltage due to sampling delay

V_P is the peak value of the common-mode voltage

f_{CM} is the common-mode frequency

As an example, to keep this error to $\frac{1}{4}$ LSB (≈ 5 mV) when operating with a 60 Hz common-mode frequency, f_{CM} , and using a 640 kHz A/D clock, f_{CLK} , would allow a peak value of the common-mode voltage, V_P , which is given by:

$$V_P = \frac{(5 \times 10^{-3})(640 \times 10^3)}{(2\pi f_{CM})(4.5)}$$

or

$$V_P = \frac{(5 \times 10^{-3})(640 \times 10^3)}{(6.28)(60)(4.5)}$$

which gives

$$V_P \approx 1.9V$$

The allowed range of analog input voltages usually places more severe restrictions on input common-mode noise levels.

An analog input voltage with a reduced span and a relatively large zero offset can be handled easily by making use of the differential input (see section 2.4 Reference Voltage).

2.3 Analog Inputs

2.3.1 Input Current

Normal Mode

Due to the internal switching action, displacement currents will flow at the analog inputs. This is due to on-chip stray capacitance to ground as shown in Figure 3.

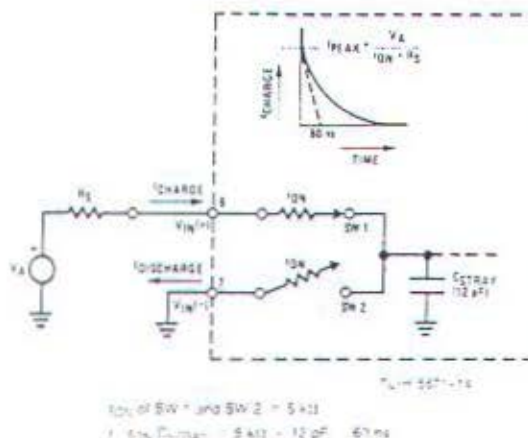


FIGURE 3. Analog Input Impedance

Functional Description [\[1\]](#)

The voltages on the capacitors during a switch event will result in currents entering the $V_{\text{sig}}(-)$ input pin and leaving the $V_{\text{sig}}(+)$ input which will depend on the analog differential input voltage levels. These current transients occur at the leading edge of the internal clocks. They rapidly decay and do not cause errors as the on-chip comparator is strobed at the next of the clock period.

8. **Final Minutes**

If the voltage source applied to the V_{IN1} or V_{IN2} pin exceeds the allowed operating range of $V_{IN} \pm 50$ mV, large input currents can flow through a parasitic diode to the V_{CC} pin. If these currents can exceed the 1 mA max. allowed

this current to the V_{CC} pin (with the current bypassed with this diode, the voltage at the $V_{IN} (+)$ pin can exceed the V_{CC} voltage by the forward voltage of this diode).

2.3.2 Input Bypass Capacitors

Bypass capacitors at the inputs will average these charges and cause a DC current to flow through the output resistances of the analog signal sources. This charge pumping action is worse for continuous conversions with the $V_{IN}(-)$ input voltage at full-scale. For continuous conversions with a 640 kHz clock frequency with the $V_{IN}(+)$ input at 5V, this DC current is at a maximum of approximately 5 μ A. Therefore, bypass capacitors should not be used at the analog inputs or the $V_{AGE}/2$ pin for high resistance sources (> 1 k Ω). If input bypass capacitors are necessary for noise filtering and high source resistance is desirable to minimize capacitor size, the detrimental effects of the voltage drop across this input resistance, which is due to the average value of the input current, can be eliminated with a full-scale adjustment; while the given source resistor and input bypass capacitor are both in place. This is possible because the average value of the input current is a precise linear function of the differential input voltage.

2.3.3 Input Source Resistance

Large values of source resistance where an input bypass capacitor is not used, will not cause errors as the input currents settle out prior to the comparison time. If a low pass filter is required in the system, use a low value of series resistor (1 to 1 k Ω) for a passive RC section or add an op amp RC active low pass filter. For low source resistance applications (1 to 1 k Ω), a 0.1 μ F bypass capacitor at the inputs will prevent noise pickup due to series lead inductance of a long wire. A 100 Ω series resistor can be used to isolate this capacitor from the ± 11 and \bar{C} and placed outside the feedback loop when the output of an op amp is used.

2.3.4 Noise

The leads to the analog inputs (pin 6 and 7) should be kept as short as possible to minimize input noise coupling. Both noise and undesired signal/clock coupling to these inputs can cause system errors. The source resistance for these inputs should, in general, be kept below 5 k Ω ; larger values of source resistance can cause undesired system noise pickup. Input bypass capacitors, placed from the analog inputs to ground, will eliminate system noise pickup but can create analog scale errors as these capacitors will average the transient input-switching currents of the A/D (see section 2.3.1). This scale error depends on both a large source

conductance and the rate of its input bypass capacitor current can be simulated by closing a half-scale adjusting the $A_1(1)$ input $V_{IN1} = 1/2$ for a proper half-scale reference voltage $2.5/2$ on Full Scale Adjustment with the same resistance and input bypass capacitor in place.

2.4 Reference Voltage

2.4.1 Signum Adjust

For magnesium applications, flexibility allows A-1111 to be designed to accommodate a $\pm V_{IN}$ or $\pm V_{IN}$ or a fixed voltage reference. This has been achieved in the design of the IC as shown in Figure 4.

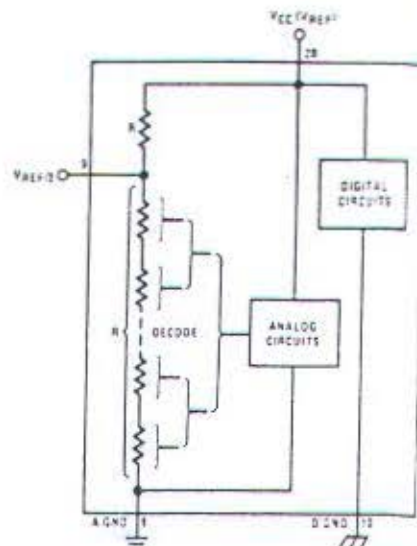
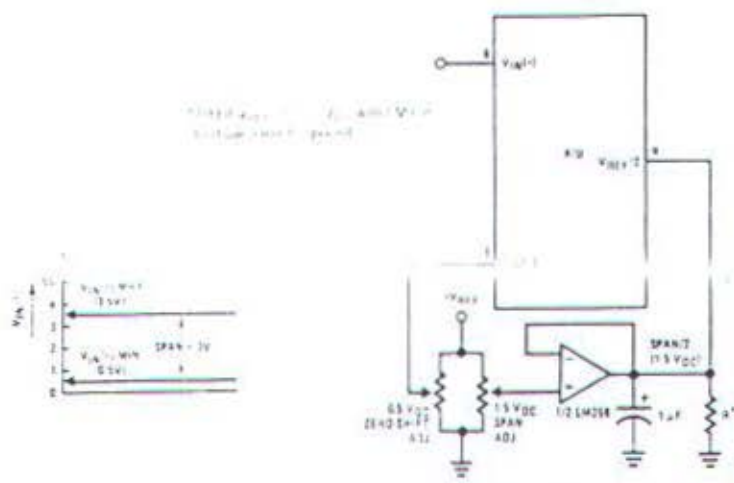


FIGURE 4. The V_{REFERENCE} Design on the I

Notice that the reference voltage for the K₁ is either the voltage applied to the V_{CC} supply pin, or a equal voltage that is externally forced at the V_{REF}/2 pin. It allows for a ratiometric voltage reference using the V_{CC} supply, a 5 V_{CC} reference voltage can be used for the supply or a voltage less than 2.5 V_{CC} can be applied. V_{REF}/2 input for increased application flexibility. Total gain to the V_{REF}/2 input is 2, making the full-scale output voltage twice the voltage at pin 9.

An example of the use of an adjusted reference voltage to accommodate a reduced span—or dynamic voltage—of the analog input voltage. If the analog input voltage is to range from 0.5 V_{DD} to 3.5 V_{DD}, instead of 0V to 5V, the span would be 3V as shown in Figure 5. With 1.5V applied to the V_{REF(-)} pin to absorb the offset, the reference voltage can be made equal to 1/2 of the 3V span or 1.5V. The A/D now will encode the V_{REF(+)} signal from 0.5V to 5V with the 0.5V input corresponding to zero and the 5V input corresponding to full-scale. The full 8 bits of resolution are therefore applied over this reduced analog input voltage range.



1000000000

Analogue Input Signal Example

- b) Accommodating an Analog Input from 0.5V (Digital Out = 00_{HEX}) to 3.5V (Digital Out = FF_{HEX})

FIGURE 5. Adapting the A/D Analog Input Voltages to Match an Arbitrary Input Signal Range

Accuracy Requirements

can be operated in a ratiometric mode or an absolute mode. In ratiometric converter applications, the reference voltage is a factor in both the source transducer and the output of the A/D; therefore, it cancels out in the final digital output. The ADC0805 is specified particularly for use in ratiometric applications with no adjustments required. In absolute applications, both the initial value and the stability of the reference voltage are important. The accuracy of the A/D converter, for $V_{REF}/2 = 4$ Vdc nominal value, initial errors of ± 10 LSB conversion errors of ± 1 LSB due to the $V_{REF}/2$ input. In reduced span applications, the stability of the $V_{REF}/2$ input voltage is more important. For example, if the span is 1 V, the analog input LSB voltage value is reduced from 20 mV (5V span) to 10 mV, and the $V_{REF}/2$ input becomes 5 mV. As can be seen, the allowed initial tolerance of the reference source correspondingly loses absolute change variations. Note that spans smaller than 1 V require tighter requirements on the initial accuracy of the reference source.

The magnitude of the reference voltage will be adjusted. Errors due to an improper adjustment appear as full-scale errors in the A/D. In IC voltage regulators may be used for reference. Ambient temperature changes are not excluded. The 2.5V IC reference mode from National has a temperature stability of ± 0.1 mV typ over 0°C to 70°C . Other temperature ranges are also available.

2.5 Errors and Reference Voltage Adjustments

2.5.1 Zero Error

The zero of the A/D does not require adjustment. If the minimum analog input voltage value, $V_{IN(MIN)}$, is not ground, a zero offset can be done. The converter can be made to output 0000 0000 digital code for this minimum input voltage by biasing the A/D $V_{IN}(-)$ input at this $V_{IN(MIN)}$ value (see Applications section). This utilizes the differential mode operation of the A/D.

The zero error of the A/D converter relates to the location of the first riser of the transfer function and can be measured by grounding the $V_{IN}(-)$ input and applying a small magnitude positive voltage to the $V_{IN}(+)$ input. Zero error is the difference between the actual DC input voltage that is necessary to just cause an output digital code transition from 0000 0000 to 0000 0001 and the ideal $1/2$ LSB value ($1/2$ LSB = 9.8 mV for $V_{REF}/2 = 2.500$ Vdc).

2.5.2 Full-Scale

The full-scale adjustment can be made by applying a differential input voltage that is $1/2$ LSB less than the desired analog full-scale voltage range and then adjusting the magnitude of the $V_{REF}/2$ input (pin 9 or the V_{CC} supply if pin 9 is not used) for a digital output code that is just changing from 1111 1110 to 1111 1111.

Functional Description (Continued)

type point analog ground that is separate from the logic ground points should be used. The power supply bypass capacitor and the self-clocking capacitor (if used) should be returned to digital ground. Any $V_{REF}/2$ bypass capacitors, analog input filter capacitors, or input signal shields should be returned to the analog ground point. A test for proper grounding is to measure the zero error of the A/D converter. Zero errors in excess of $1/2$ LSB can usually be attributed to improper board layout and wiring (see section 8 for measuring the zero error).

TESTING THE A/D CONVERTER

Several methods of testing are possible with this converter. The most common method is to apply an analog input voltage to the converter and use LEDs to display the resulting digital output code as shown in Figure 7.

For ease of testing, the $V_{REF}/2$ (pin 9) should be supplied 1.560 V_{DC} and a V_{DD} supply voltage of 5.12 V_{DC} be used. This provides an LSB value of 20 mV.

For scale adjustment to be made, an analog input of 5.090 V_{DC} ($5.120 - 1/2$ LSB) should be applied to pin 1 with the $V_{REF}/2$ pin grounded. The value of $V_{REF}/2$ input voltage should then be adjusted until the output code is just changing from 1111 1110 to 1111 1111. This value of $V_{REF}/2$ should then be used for all the

digital output LED display can be decoded by dividing bits into 2 hex characters, the 4 most significant (MS) and 4 least significant (LS). Table 1 shows the fractional equivalent of these two 4-bit groups. By adding the values obtained from the "VMS" and "VLS" columns in Table 1, the nominal value of the digital display (when

$V_{REF}/2 = 2.560$ V) can be determined. For example, for an output LED display of 1011 0110 or B6 (in hex), the voltage values from the table are $3.520 + 0.120$ or 3.640 V_{DC}. These voltage values represent the center-values of a perfect A/D converter. The effects of quantization error have to be accounted for in the interpretation of the test results.

For a higher speed test system, or to obtain plotted data, a digital-to-analog converter is needed for the test set-up. An accurate 10-bit DAC can serve as the precision voltage source for the A/D. Errors of the A/D under test can be expressed as either analog voltages or differences in 2 digital words.

A useful A/D tester that uses a DAC and provides the error as an analog output voltage is shown in Figure 8. The 2 op amps can be eliminated if a lab DVM with a numerical subtraction feature is available to read the difference voltage, "A-C", directly. The analog input voltage can be supplied by a low frequency ramp generator and an X-Y plotter can be used to provide analog error (Y axis) versus analog input (X axis).

For operation with a microprocessor or a computer-based test system, it is more convenient to present the errors digitally. This can be done with the circuit of Figure 9, where the output code transitions can be detected as the 10-bit DAC is incremented. This provides $1/4$ LSB steps for the 8-bit A/D under test. If the results of this test are automatically plotted with the analog input on the X axis and the error (in LSB's) as the Y axis, a useful transfer function of the A/D under test results. For acceptance testing, the plot is not necessary and the testing speed can be increased by establishing internal limits on the allowed error for each code.

4.0 MICROPROCESSOR INTERFACING

To discuss the interface with 8080A and 6800 microprocessors, a common sample subroutine structure is used. The microprocessor starts the A/D, reads and stores the results of 16 successive conversions, then returns to the user's program. The 16 data bytes are stored in 16 successive memory locations. All Data and Addresses will be given in hexadecimal form. Software and hardware details are provided separately for each type of microprocessor.

4.1 Interfacing 8080 Microprocessor Derivatives (8048, 8085)

This converter has been designed to directly interface with derivatives of the 8080 microprocessor. The A/D can be mapped into memory space (using standard memory address decoding for CS and the MEMR and MEMW strobes) or it can be controlled as an I/O device by using the I/OH and I/OV strobes and decoding the address bits A0 → A7 (or address bits A8 → A15 as they will contain the same 8-bit address information) to obtain the CS input. Using the I/O space provides 256 additional addresses and may allow a simpler 8-bit address decoder but the data can only be input to the accumulator. To make use of the additional memory reference instructions, the A/D should be mapped into memory space. An example of an A/D in I/O space is shown in Figure 10.

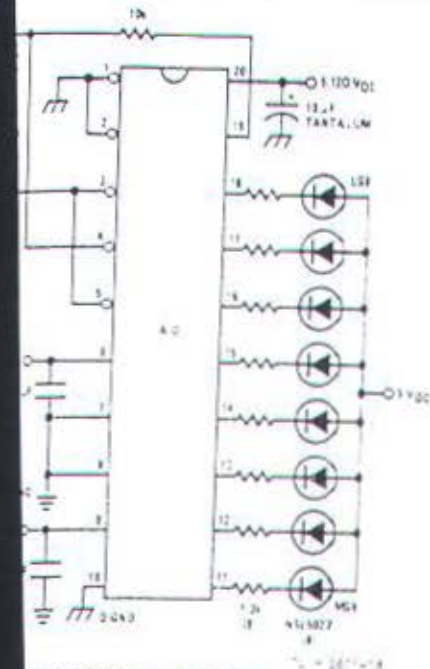


FIGURE 7. Basic A/D Tester

ADC0801/ADC0802/ADC0803/ADC0804/ADC0805

Functional Description (Continued)

Fun

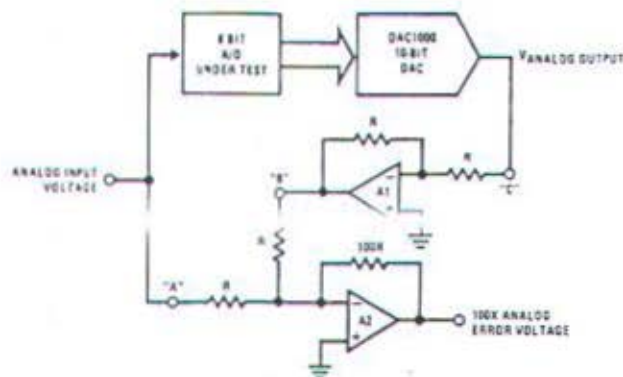


FIGURE 8. A/D Tester with Analog Error Output



FIGURE 9. Basic "Digital" A/D Tester

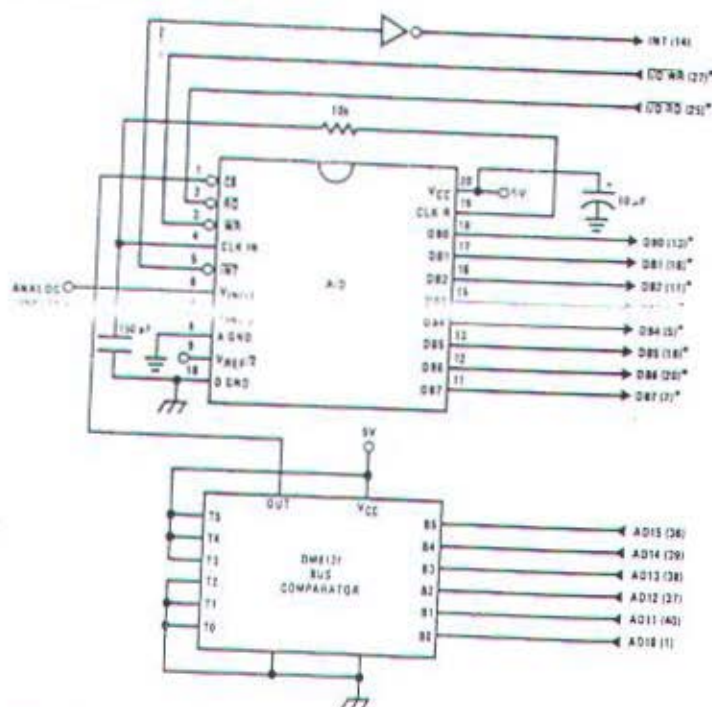
TL/H/5671-19

TABLE I. DECODING THE DIGITAL OUTPUT LEDs

| HEX | BINARY | FRACTIONAL BINARY VALUE FOR | | OUTPUT VOLTAGE CENTER VALUES WITH $V_{REF}/2 = 2.560 V_{DC}$ | |
|-----|---------|-----------------------------|----------|--|------------|
| | | MS GROUP | LS GROUP | VMS GROUP* | VLS GROUP* |
| F | 1 1 1 1 | 15/16 | 15/256 | 4.800 | 0.300 |
| E | 1 1 1 0 | 7/8 | 7/128 | 4.480 | 0.280 |
| D | 1 1 0 1 | 13/16 | 13/256 | 4.160 | 0.260 |
| C | 1 1 0 0 | 3/4 | 3/64 | 3.840 | 0.240 |
| B | 1 0 1 1 | 11/16 | 11/256 | 3.520 | 0.220 |
| A | 1 0 1 0 | 5/8 | 5/128 | 3.200 | 0.200 |
| 9 | 1 0 0 1 | 9/16 | 9/256 | 2.880 | 0.180 |
| 8 | 1 0 0 0 | 1/2 | 1/32 | 2.560 | 0.160 |
| 7 | 0 1 1 1 | 7/16 | 7/256 | 2.240 | 0.140 |
| 6 | 0 1 1 0 | 3/8 | 3/128 | 1.920 | 0.120 |
| 5 | 0 1 0 1 | 5/16 | 5/256 | 1.600 | 0.100 |
| 4 | 0 1 0 0 | 1/4 | 1/64 | 1.280 | 0.080 |
| 3 | 0 0 1 1 | 3/16 | 3/256 | 0.960 | 0.060 |
| 2 | 0 0 1 0 | 1/8 | 1/128 | 0.640 | 0.040 |
| 1 | 0 0 0 1 | 1/16 | 1/256 | 0.320 | 0.020 |
| 0 | 0 0 0 0 | | | 0 | 0 |

*Display Output = VMS Group + VLS Group

Functional Description (Continued)



Note 1: Pin numbers for the DP8228 system controller, others are INS8080A.
 Note 2: Pin 20 of the INS8228 must be tied to +12V through a 1 kΩ resistor to generate the RST 7 instruction when an interrupt is acknowledged as required by the accompanying sample program.

FIGURE 10. ADC0801-INS8080A CPU Interface

SAMPLE PROGRAM FOR FIGURE 10 ADC0801-INS8080A CPU INTERFACE

```

C3 00 03  RST 7:      JNP    LD DATA
.
.
21 00 02  START:      LXI H 0200H      ;HL pair will point to
                                ;data storage locations
31 00 04  RETURN:      LXI SP 0400H    ;Initialize stack pointer (Note 1)
7D                MOV A, L            ;Test # of bytes entered
FE 0F                CPI 0FH          ;If # = 16, JMP to
CA 13 01            JZ CONT          ;user program
D3 E0                OUT EOH         ;Start A/D
FB                EI                ;Enable interrupt
00                NOP                ;Loop until end of
C3 CF 01            JMP LOOP         ;conversion
.
.
.
CONT:          .
.
.
.
.
.
.
.
DB E0  LD DATA:      IN EOH          ;Load data into accumulator
77                MOV M, A           ;Store data
23                INX H              ;Increment storage pointer
C3 C3 01            JMP RETURN
    
```

Stack pointer must be dimensioned because a RST 7 instruction pushes the PC onto the stack.
 Address used were arbitrarily chosen.

Functional Description (Continued)

The standard control bus signals of the 8080 CS, RD and WR can be directly wired to the digital control inputs of the A/D and the bus timing requirements are met to allow both starting the converter and outputting the data onto the data bus. A bus driver should be used for larger microprocessor systems where the data bus leaves the PC board and/or must drive capacitive loads larger than 100 pF.

4.1.1 Sample 8080A CPU Interfacing Circuitry and Program

The following sample program and associated hardware shown in Figure 11 may be used to input data from the converter to the INS8080A CPU chip set (comprised of the INS8080A microprocessor, the INS8228 system controller and the INS8224 clock generator). For simplicity, the A/D is controlled as an I/O device, specifically an 8-bit bi-directional port located at an arbitrarily chosen port address, E0. The TRI-STATE output capability of the A/D eliminates the need for a peripheral interface device, however address decoding is still required to generate the appropriate CS for the converter.

It is important to note that in systems where the A/D converter is 1-of-8 or less I/O mapped devices, no address decoding circuitry is necessary. Each of the 8 address bits (A0 to A7) can be directly used as CS inputs—one for each I/O device.

4.1.2 INS8048 Interface

The INS8048 interface technique with the ADC0801 series (see Figure 11) is simpler than the 8080A CPU interface. There are 24 I/O lines and three test input lines in the 8048.

0 of port 1) is used as the chip select signal to the A/D, thus eliminating the use of an external address decoder. Bus control signals RD, WR and INT of the 8048 are tied directly to the A/D. The 16 converted data words are stored at on-chip RAM locations from 20 to 2F (Hex). The RD and WR signals are generated by reading from and writing into a dummy address, respectively. A sample interface program is shown below.

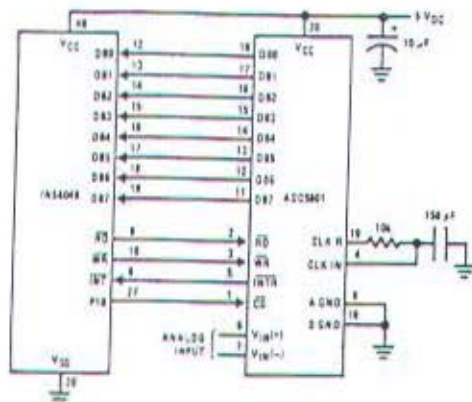


FIGURE 11. INS8048 Interface
SAMPLE PROGRAM FOR FIGURE 11 INS8048 INTERFACE

```

04 10      JMP      10H          ; Program starts at addr 10
04 30      ORG      3H          ;
04 50      JMP      50H          ; Interrupt jump vector
04 70      ORG      10H          ; Main program
04 7E      ANDL     P1, #0FFH    ; Chip select
04 81      MOVX     A, @R1        ; Read in the 1st data
                                ; to reset the intr
04 81      START:  DEL      P1, #1 ; Set port pin high
04 82      MOV      R0, #20H      ; Data address
04 83      MOV      R1, #0FFH      ; Dummy address
04 84      MOV      R2, #10H      ; Counter for 16 bytes
04 85      AGAIN:  MOV      A, #0FFH ; Set ACC for intr loop
04 86      ANDL     P1, #0FFH      ; Send CS (bit 0 of P1)
04 87      MOVX     @R1, A         ; Send WR out
04 88      EN      I              ; Enable interrupt
04 89      LOOP:   JPE      LOOP    ; Wait for interrupt
04 90      DJNZ     R2, AGAIN      ; If 16 bytes are read
04 91      NOP      ; go to user's program
04 92      NOP
04 93      ORG      50H
04 94      MOVX     A, @R1        ; Input data, CS still low
04 95      MOV      @R0, A        ; Store in memory
04 96      INC      R0            ; Increment storage counter
04 97      ORL      P1, #1        ; Reset CS signal
04 98      CLR      A            ; Clear ACC to get out of
04 99      RETR              ; the interrupt loop

```

Func

4.2 Interf

The Z-80
word. On
sally more
nals are t
load slot
vantage
that the C
and WR

A/D in I/O

FIG1

Additional
are availa
for which
during I/O
selection
ing mode.

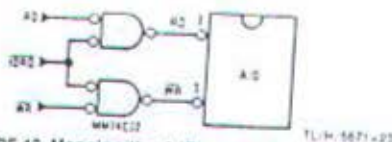
4.3 Interf (6502

The conti
does not i
boys a si
be deliver
mapped li
ndicates t
an interfac
the 6800
using 1/2 C

Functional Description (Continued)

Interfacing the Z-80

The Z-80 control bus is slightly different from that of the 8080. General RD and WR strobes are provided and separate memory request, MREQ, and I/O request, IORQ, signals are used which have to be combined with the generalized strobes to provide the equivalent 8080 signals. An advantage of operating the A/D in I/O space with the Z-80 is that the CPU will automatically insert one wait state (the RD and WR strobes are extended one clock period) to allow more time for the I/O devices to respond. Logic to map the A/D in I/O space is shown in Figure 13.



SAMPLE PROGRAM FOR FIGURE 14 ADC0801-MC6800 CPU INTERFACE

Note 1: In order for the microprocessor to service subroutines and interrupts, the stack pointer must be dimensioned in the user's program.

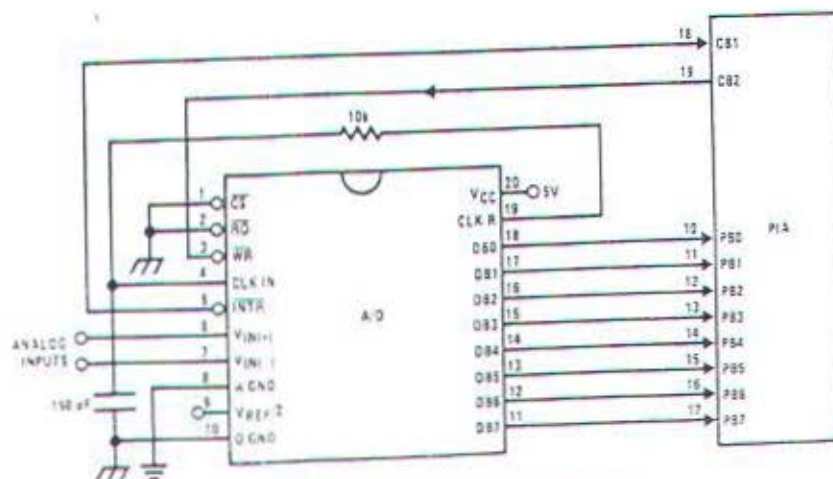


FIGURE 15. ADC0801-MC6820 PIA Interface

Functional Description (Continued)

SAMPLE PROGRAM FOR FIGURE 15 ADC0801-MC6820 PIA INTERFACE

```

0010  CE 00 38      DATAIN  LDX      #30038      ; Upon IRQ low CPU
0013  FF FF FF      STX      $FFFFH      ; jumps to 0038
0016  F6 FF 06      LDAA     PIAORH      ; Clear possible IRQ flags
0019  4F           PIABA
001A  B7 00 07      STAA     PIACRB      ; Set Port B as input
001D  B7 80 06      STAA     PIAORB
0020  05           CLI
0021  C6 34      LDAB     #534
;
0025  F7 80 07      CONVRT  STAB     PIACRB      ; Starts ADC0801
0028  B7 80 07      STAA     PIACRB
002B  3E           WAI      ; Wait for interrupt
002C  DE 40      LDX      TEMP1
002E  8C 02 0F      CPX      #50207      ; Is final data stored?
0031  27 0F      BEQ      ENDP
0033  08           INX
0034  DF 40      STX      TEMP1
0036  20 E0      BRA      CONVRT
0038  DE 40      INTERPT  LDX      TEMP1
003A  B6 90 06      LDAA     PIAORB      ; Read data in
003D  A7 0C      STAA     X      ; Store it at X
003F  35           RTI
0040  02 00      TEMP1  FDB     $0200      ; Starting address for
; data storage
0042  C6 02 00      ENDP   LDX      #50200      ; Reinitialize TEMP1
0045  DF 40      STX      TEMP1
0047  39           RTS      ; Return from subroutine
; To user's program
PIACRB  EQU      $8006
PIACRB  EQU      $8007

```

Following schematic and sample subroutine (DATA IN) be used to interface (up to) 8 ADC0801's directly to the 6800 CPU. This scheme can easily be extended to allow interface of more converters. In this configuration the converters are (arbitrarily) located at HEX address 5000 in MC6820 memory space. To save components, the signal is derived from just one RC pair on the first converter. This output drives the other A/Ds.

The converters are started simultaneously with a STORE instruction at HEX address 5000. Note that any other HEX address of the form 5XXX will be decoded by the circuit, keeping all the CS inputs low. This can easily be avoided by a more definitive address decoding scheme. All the outputs are ORed together to insure that all A/Ds have completed their conversion before the microprocessor is interrupted.

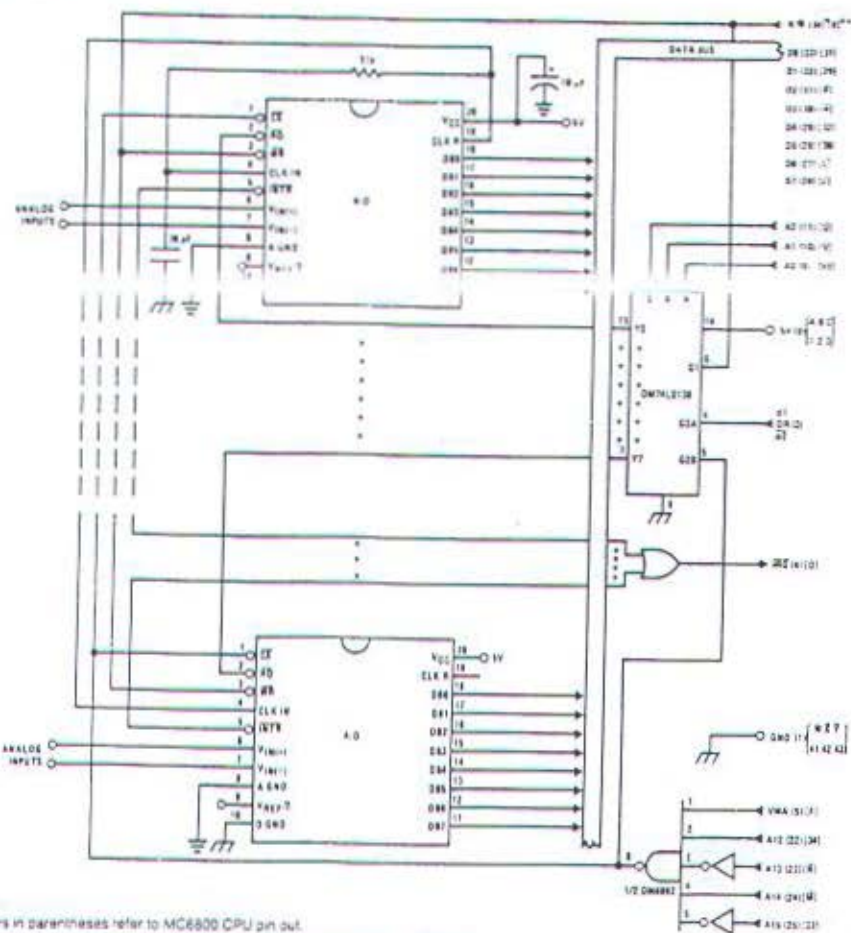
Subroutine, DATA IN, may be called from anywhere in user's program. Once called, this routine initializes the

CPU, starts all the converters simultaneously and waits for the interrupt signal. Upon receiving the interrupt, it reads the converters (from HEX addresses 5000 through 5007) and stores the data successively at (arbitrarily chosen) HEX addresses 0200 to 0207, before returning to the user's program. All CPU registers then recover the original data they had before servicing DATA IN.

5.2 Auto-Zeroed Differential Transducer Amplifier and A/D Converter

The differential inputs of the ADC0801 series eliminate the need to perform a differential to single ended conversion for a differential transducer. Thus, one op amp can be eliminated since the differential to single ended conversion is provided by the differential input of the ADC0801 series. In general, a transducer preamp is required to take advantage of the full A/D converter input dynamic range.

Functional Description (Continued)



- 1: Numbers in parentheses refer to MC6800 CPU pin out.
2: Numbers of letters in brackets refer to standard M6800 system common bus code.

FIGURE 16. Interfacing Multiple A/Ds in an MC6800 System
SAMPLE PROGRAM FOR FIGURE 16 INTERFACING MULTIPLE A/Ds IN AN MC6800 SYSTEM

| ADDRESS | HEX CODE | MNEMONICS | COMMENTS |
|---------|----------|-----------------|------------------------------|
| 0010 | DF 44 | DATAIN STX TEMP | : Save Contents of X |
| 0012 | CE 00 2A | LDX #002A | : Upon IRQ LOW CPU |
| 0015 | FF FF 78 | STX \$FFFF | : Jumps to 002A |
| 0018 | 97 80 00 | STAA \$8000 | : Starts all A/D's |
| 001B | 0E | CLI | |
| 001C | 3E | WAI | : Wait for interrupt |
| 001D | CE 50 40 | LDX #5000 | |
| 0020 | DF 40 | STX INDEX1 | : Reset both INDEX |
| 0022 | CE 00 00 | LDX #0000 | : 1 and 2 to starting |
| 0025 | FF 40 | STX INDEX2 | : addresses |
| 0027 | 78 44 | LDX TEMP | |
| 0028 | 39 | RTS | : Return from Subroutine |
| 002A | CE 40 | LDX INDEX1 | : INDEX1 → X |
| 002C | A6 00 | LDAA X | : Read data in from A/D at X |
| 002E | 08 | INX | : Increment X by one |
| 002F | DF 40 | STX INDEX1 | : X → INDEX1 |
| 0031 | DE 42 | LDX INDEX2 | : INDEX2 → X |

Functional C

| ADDRESS |
|---------|
| 0015 |
| 0018 |
| 0038 |
| 003A |
| 003B |
| 003D |
| 003F |
| 0040 |
| 0042 |
| 0044 |

Note: In order for

For amplification
the input offset
preamp. Figure
whose offset vo
subroutine which
error system. T
for this preamp
obviously require
the differential i

V_0

(V_{OS2})

where i_x is the
error terms of
 $V_{OS2} - V_0$
scheme.
The INS808
variable Pe
ng and input
the PPI is)
Ain Port A
output ports
close the

tional Description (Continued)

SAMPLE PROGRAM FOR FIGURE 16 INTERFACING MULTIPLE A/Ds IN AN MC6800 SYSTEM

| ADDRESS | HEX CODE | MNEMONICS | COMMENTS |
|---------|----------|-------------------|-------------------------------------|
| 33 | A7 00 | STAA X | ; Store data at X |
| 35 | AC 02 07 | CMX #0207 | ; Have all A/D's been read? |
| 36 | 27 05 | BEQ RETURN | ; Yes: branch to RETURN |
| 3A | 08 | INX | ; No: increment X by one |
| 3B | DF 42 | STX INDEX2 | ; X → INDEX2 |
| 3D | 20 E8 | BRA INTRPT | ; Branch to 002A |
| 3F | 38 | RETURN RTI | |
| 40 | 50 00 | INDEX1 FDB \$0000 | ; Starting address for A/D's |
| 42 | 02 00 | INDEX2 FDB \$0000 | ; Starting address for data storage |
| 44 | 00 00 | END | |

order for the microprocessor to service subroutines and interrupts, the stack pointer must be dimensioned in the user's program

location of DC input signals, a major system error is the offset voltage of the amplifiers used for the Figure 17 is a gain of 100 differential preamp. Offset voltage errors will be cancelled by a zeroing operation which is performed by the INS8080A microprocessor. The total allowable input offset voltage error preamp is only 50 μ V for 1/4 LSB error. This would require very precise amplifiers. The expression for differential output voltage of the preamp is:

$$V_O = \underbrace{[V_{IN}(-) - V_{IN}(+)]}_{\text{SIGNAL}} \underbrace{\left[1 + \frac{2R_2}{R_1}\right]}_{\text{GAIN}} + \underbrace{[V_{OS1} - V_{OS2}]}_{\text{DC ERROR TERM}} \underbrace{\left[1 + \frac{2R_2}{R_1}\right]}_{\text{GAIN}}$$

is the current through resistor R_X . All of the offset can be cancelled by making $\pm I_X R_X = V_{OS1} = -V_{OS2}$. This is the principle of this auto-zeroing

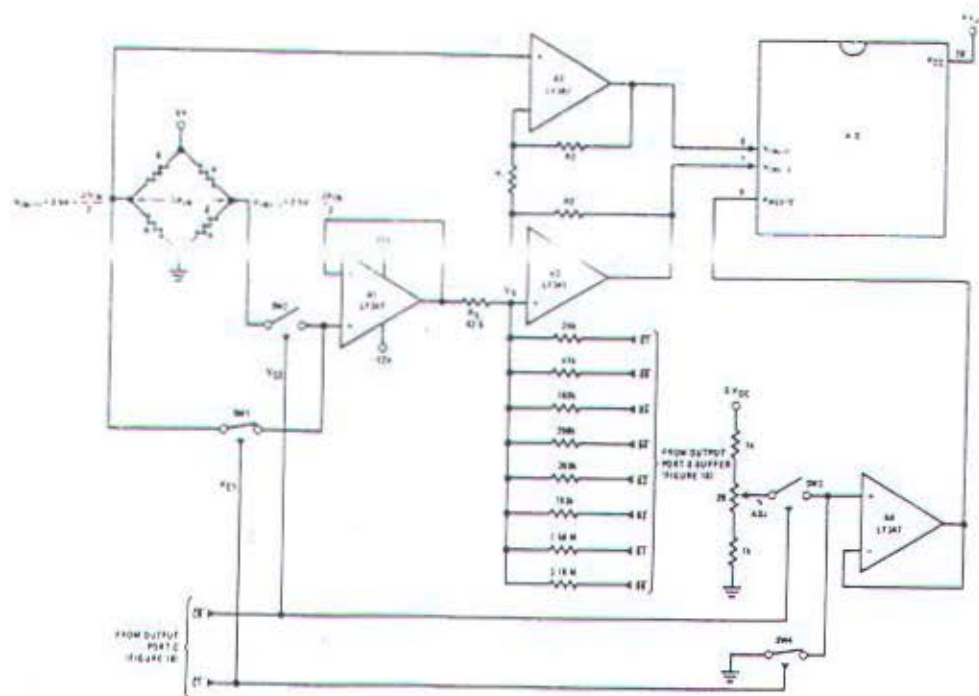
680A uses the 3 I/O ports of an INS8255 Peripheral Interface (PI) to control the auto-zeroing data from the ADC0801 as shown in Figure 18. programmed for basic I/O operation (mode 0) A being an input port and Ports B and C being outputs. Two bits of Port C are used to alternately open the 2 switches at the input of the preamp. Switch

SW1 is closed to force the preamp's differential input to be zero during the zeroing subroutine and then opened and SW2 is then closed for conversion of the actual differential input signal. Using 2 switches in this manner eliminates concern for the ON resistance of the switches as they must conduct only the input bias current of the input amplifiers.

Output Port B is used as a successive approximation register by the 6080 and the binary scaled resistors in series with each output bit create a D/A converter. During the zeroing subroutine, the voltage at V_X increases or decreases as required to make the differential output voltage equal to zero. This is accomplished by ensuring that the voltage at the output of A1 is approximately 2.5V so that a logic "1" (5V) on any output of Port B will source current into node V_X thus raising the voltage at V_X and making the output differential more negative. Conversely, a logic "0" (0V) will pull current out of node V_X and decrease the voltage, causing the differential output to become more positive. For the resistor values shown, V_X can move ± 12 mV with a resolution of 50 μ V, which will null the offset error term to 1/4 LSB of full-scale for the ADC0801. It is important that the voltage levels that drive the auto-zero resistors be constant. Also, for symmetry, a logic swing of 0V to 5V is convenient. To achieve this, a CMOS buffer is used for the logic output signals of Port B and this CMOS package is powered with a stable 5V source. Buffer amplifier A1 is necessary so that it can source or sink the D/A output current.

ADC0801/ADC0802/ADC0803 ADC0804/ADC0805

Functional Description (Continued)



Note 1: R2 = 45.5 A1

Note 2: Switches are LMC13334 CMOS analog switches.

Note 3: The 9 resistors used in the auto-zero section can be $\pm 5\%$ tolerance

FIGURE 17. Gain of 100 Differential Transducer Preamp

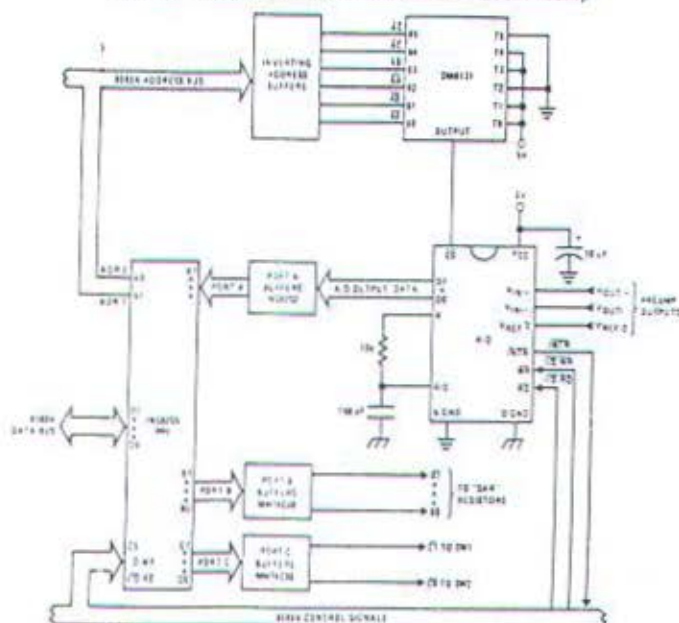


FIGURE 18. Microprocessor Interface Circuitry for Differential Preamp

T.L.W./5871-3

Port A
Port B
Port C
API or
Progr
ackno

5.3 Mul Driv

in data 1
varter (i
gram e)
need for
ing. Fig
of data
pleted i
interrupt
any sec
convert
A-D 2:
lowest
address
the key
type fil
the pro
this su
partial
bits of
interrupt
status
ing por
which
the res
the del
data of

Flow chart for the zeroing subroutine is shown in Figure 19. It must be noted that the ADC0801 series will output a zero code when it converts a negative input [$V_{IN}(-) > V_{IN}(+)$]. Also, a logic inversion exists as all of the I/O ports are buffered with inverting gates.

If the data read is zero, the differential output voltage is negative, so a bit in Port B is cleared to pull V_X more positive which will make the output more positive for the next conversion. If the data read is not zero, the output voltage is positive so a bit in Port B is set to make V_X more negative and the output more negative. This continues for 8 approximations and the differential output eventually converges to within 5 mV of zero.

are compatible with the BUC 80/10 microcomputer system. In particular:
 Port A and the ADC0801 are at port address E4
 Port B is at port address E5
 Port C is at port address E6
 Port control word port is at port address E7
 Program Counter automatically goes to ADDR:3C3D upon acknowledgement of an interrupt from the ADC0801

Multiple A/D Converters in a Z-80 Interrupt Driven Mode

In acquisition systems where more than one A/D converter (or other peripheral device) will be interrupting proper execution of a microprocessor, there is obviously a need for the CPU to determine which device requires service. Figure 27 and the accompanying software is a method of determining which of 7 ADC0801 converters has completed a conversion (INTR asserted) and is requesting an output. This circuit allows starting the A/D converters in sequence, but will input and store valid data from the converters with a priority sequence of A/D 1 being read first, second, etc., through A/D 7 which would have the lowest priority for data being read. Only the converter's INTR is asserted will be read.

As to decoding circuitry is the DM74LS373, 8-bit D flip-flop. When the Z-80 acknowledges the interrupt, program is vectored to a data input Z-80 subroutine. This routine will read a peripheral status word from the 373 which contains the logic state of the INTR output of the converters. Each converter which initiates an output will place a logic "0" in a unique bit position in the word and the subroutine will determine the identity of the converter and execute a data read. An identifier word indicates which A/D the data came from is stored in sequential memory location above the location of the data so the program can keep track of the identity of the converter.

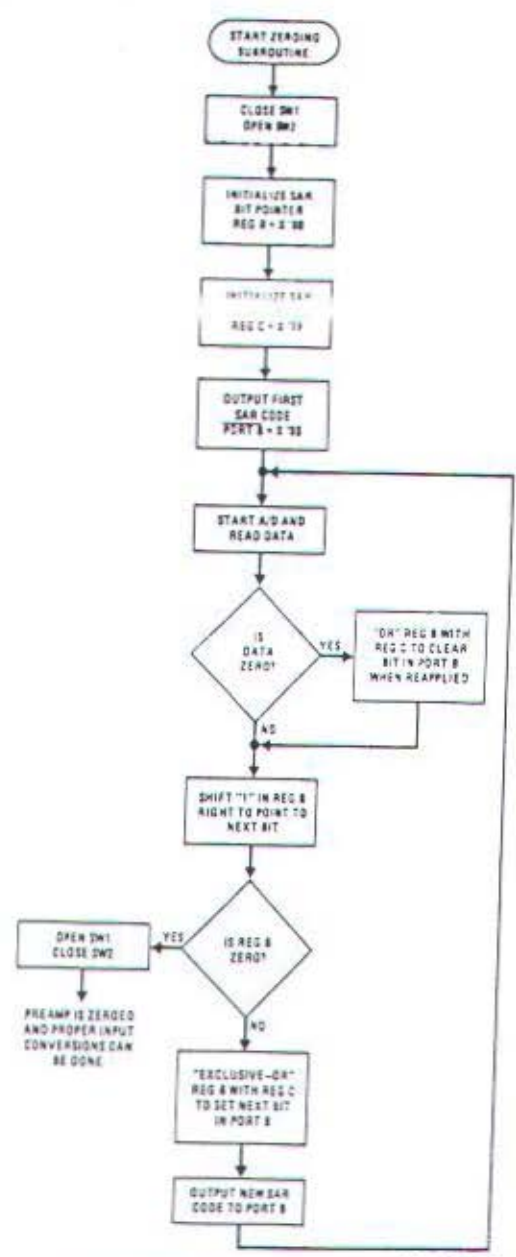


FIGURE 19. Flow Chart for Auto-Zero Routine

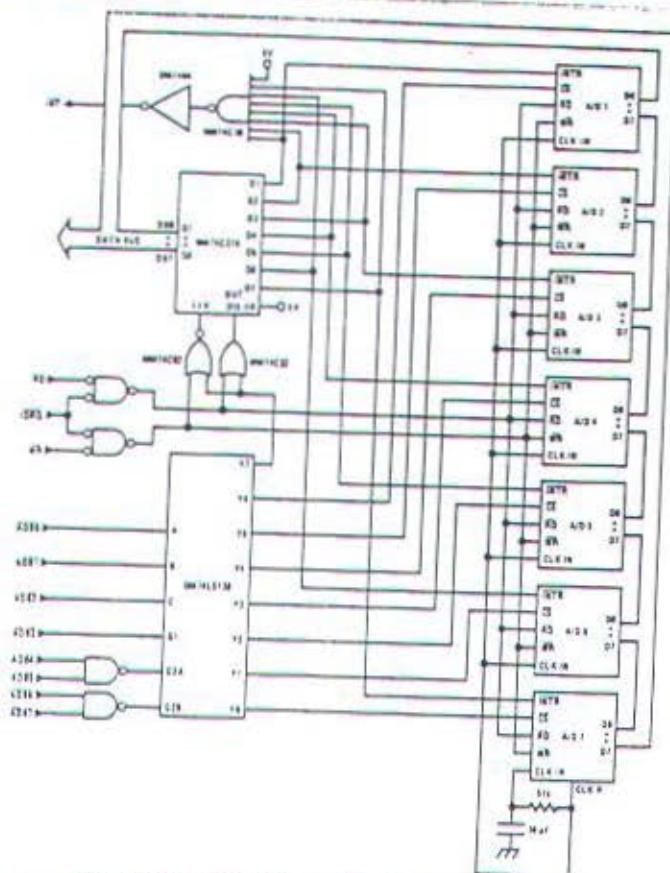


FIGURE 21. Multiple A/Ds with Z-80 Type Microprocessor

TL/H/5671-29

INTERRUPT SERVICING SUBROUTINE

| C | OBJ CODE | SOURCE STATEMENT | COMMENT |
|----|----------|------------------|--|
| 03 | 55 | PUSH HL | : Save contents of all registers affected by |
| 08 | C5 | PUSH BC | : this subroutine. |
| 0A | F5 | PUSH AF | : Assumed INT mode 1 earlier set. |
| 0E | 21 00 3E | LD (HL), X3E00 | : Initialize memory pointer where data will be stored. |
| 12 | 03 C1 | LD C, X01 | : C register will be port ADDR of A/D converters. |
| 16 | D3 00 | OUT X00, A | : Load peripheral status word into 8-bit latch. |
| 1A | D3 00 | IN A, X00 | : Load status word into accumulator. |
| 1E | 47 | LD B, A | : Save the status word. |
| 22 | 79 | TEST LD A, C | : Test to see if the status of all A/D's have |
| 26 | FE 08 | CF, X0B | : been checked. If so, exit subroutine |
| 2A | CA 60 00 | JPL, DONE | |
| 2E | 78 | LD A, B | : Test a single bit in status word by looking for |
| 32 | 1F | RFA | : a '1' to be rotated into the CARRY (an INT |
| 36 | 47 | LD B, A | : is loaded as a '1'). If CARRY is set then load |
| 3A | DA 55 00 | JFC, LOAD | : contents of A/D at port ADDR in C register. |
| 3E | 0C | INCC | : If CARRY is not set, increment C register to point |
| 42 | C3 45 00 | JP, TEST | : to next A/D, then test next bit in status word. |
| 46 | ED 78 | LOAD IN A, (C) | : Read data from interrupting A/D and invert |
| 4A | EE FF | XOR FF | : the data. |
| 4E | 77 | LD (HL), A | : Store the data |
| 52 | 2C | INCL | |
| 56 | 71 | LD (HL), C | : Store A/D identifier (A/D port ADDR). |
| 5A | 2C | INCL | |
| 5E | C3 51 00 | JP, NEXT | : Test next bit in status word. |
| 62 | F1 | POP AF | : Re-establish all registers as they were |
| 66 | C1 | POP BC | : before the interrupt. |
| 6A | E1 | POP HL | |
| 6E | C9 | RET | : Return to original program |

General Description

The LM136/LM236 and LM336 integrated circuits are precision 2.5V shunt regulator diodes. These monolithic $\pm 0.1\%$ voltage references operate as a low temperature coefficient 2.5V zener with 0.2Ω dynamic impedance. A third terminal on the LM136 allows the reference voltage and temperature coefficient to be trimmed easily.

The LM136 series is useful as a precision 2.5V low voltage reference for digital voltmeters, power supplies or op amp circuitry. The 2.5V make it convenient to obtain a stable reference from 5V logic supplies. Further, since the LM136 operates as a shunt regulator, it can be used as either a positive or negative voltage reference.

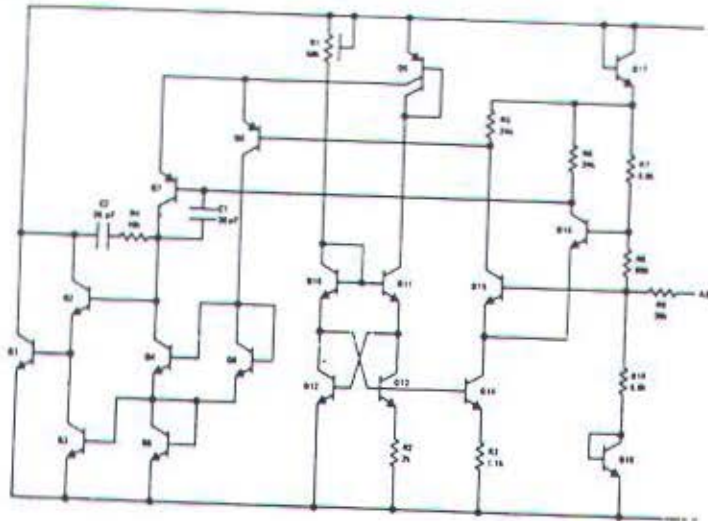
The LM136 is rated for operation over -55°C to $+125^\circ\text{C}$ while the LM236 is rated over a -25°C to $+85^\circ\text{C}$

temperature range. Both are packaged in a TO-46 package. The LM336 is rated for operation over a 0°C to $+70^\circ\text{C}$ temperature range and is available in either a three lead TO-46 package or a TO-92 plastic package.

Features

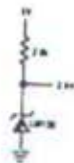
- Low temperature coefficient
- Wide operating current of $300\mu\text{A}$ to 10mA
- 0.2Ω dynamic impedance
- $\pm 1\%$ initial tolerance available
- Guaranteed temperature stability
- Easily trimmed for minimum temperature drift
- Fast turn-on
- Three lead transistor package

Schematic Diagram

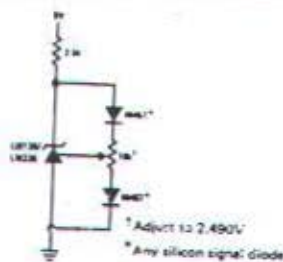


Typical Applications

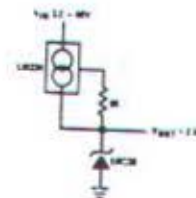
2.5V Reference

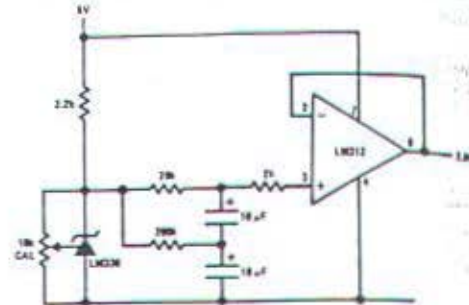
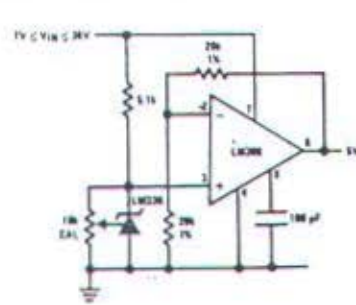


2.5V Reference with Minimum Temperature Coefficient



Wide Input Range Reference





TO-92
Plastic Package



TO-18
Metal Can Package



General

The LM136/LM236/LM336 are precision 2.5V reference diodes with a temperature coefficient of 0.01%/°C. The LM136/LM236/LM336 are available in TO-92, TO-18, and TO-25 packages. The LM136/LM236/LM336 are available in both silicon and germanium. The LM136/LM236/LM336 are available in both 2.5V and 5.0V reference voltages. The LM136/LM236/LM336 are available in both 100mA and 200mA current ratings. The LM136/LM236/LM336 are available in both 100mA and 200mA current ratings.

Schematic

Typical

General Description

The LM136-5.0/LM236-5.0/LM336-5.0 integrated circuits are precision 5.0V shunt regulator diodes. These monolithic IC voltage references operate as a low temperature coefficient 5.0V zener with 0.6Ω dynamic impedance. A third terminal on the LM136-5.0 allows the reference voltage and temperature coefficient to be trimmed easily.

The LM136-5.0 series is useful as a precision 5.0V low voltage reference for digital voltmeters, power supplies or op amp circuitry. The 5.0V make it convenient to obtain a stable reference from low voltage supplies. Further, since the LM136-5.0 operates as a shunt regulator, it can be used as either a positive or negative voltage reference.

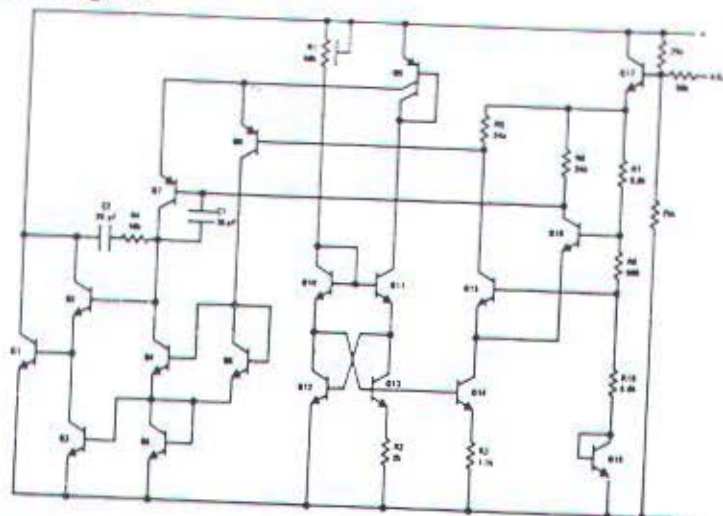
The LM136-5.0 is rated for operation over -55°C to $+125^{\circ}\text{C}$ while the LM236-5.0 is rated over a -25°C to $+85^{\circ}\text{C}$ temperature range. Both are packaged in a TO-46

package. The LM336-5.0 is rated for operation over a 0°C to $+70^{\circ}\text{C}$ temperature range and is available in either a three lead TO-46 package or a TO-92 plastic package. For applications requiring 2.5V see LM136-2.5.

Features

- Adjustable 4V to 6V
- Low temperature coefficient
- Wide operating current of 400 μA to 10 mA
- 0.6Ω dynamic impedance
- $\pm 1\%$ initial tolerance available
- Guaranteed temperature stability
- Easily trimmed for minimum temperature drift
- Fast turn-on
- Three lead transistor package

Schematic Diagram



Typical Applications

5.0V Reference

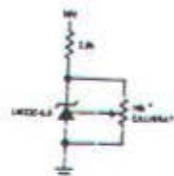


† Adjust to 5.00V
* Any silicon signal diode

5.0V Reference with Minimum Temperature Coefficient



Trimmed 4V to 6V Reference with Temperature Coefficient Independent of Breakdown Voltage



* Does not affect temperature coefficient

General Description

The LM135 series are precision, easily-calibrated, integrated circuit temperature sensors. Operating as a 2-terminal zener, the LM135 has a breakdown voltage directly proportional to absolute temperature at $+10 \text{ mV}/^\circ\text{K}$. With less than 1Ω dynamic impedance the device operates over a current range of $400 \mu\text{A}$ to 5 mA with virtually no change in performance. When calibrated at 25°C the LM135 has typically less than 1°C error over a 100°C temperature range. Unlike other sensors the LM135 has a linear output.

Applications for the LM135 include almost any type of temperature sensing over a -55°C to $+150^\circ\text{C}$ temperature range. The low impedance and linear output make interfacing to readout or control circuitry especially easy.

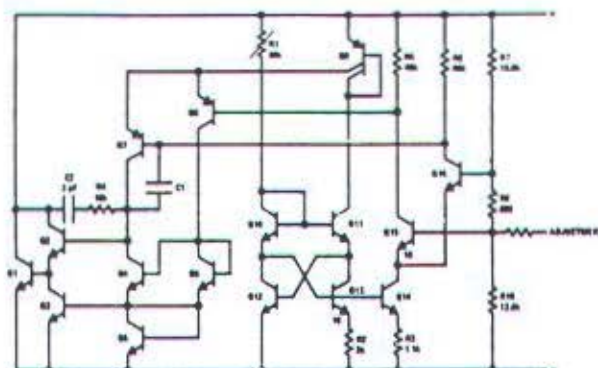
The LM135 operates over a -55°C to $+150^\circ\text{C}$ temperature range while the LM235 operates over a -40°C

to $+125^\circ\text{C}$ temperature range. The LM335 operates from -40°C to $+100^\circ\text{C}$. The LM135/LM235/LM335 are available packaged in hermetic TO-46 transistor packages while the LM335 is also available in plastic TO-92 packages.

Features

- Directly calibrated in $^\circ\text{Kelvin}$
- 1°C initial accuracy available
- Operates from $400 \mu\text{A}$ to 5 mA
- Less than 1Ω dynamic impedance
- Easily calibrated
- Wide operating temperature range
- 200°C overrange
- Low cost

Schematic Diagram



Typical Applications

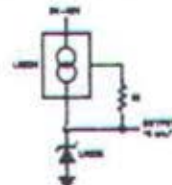
Basic Temperature Sensor



Calibrated Sensor

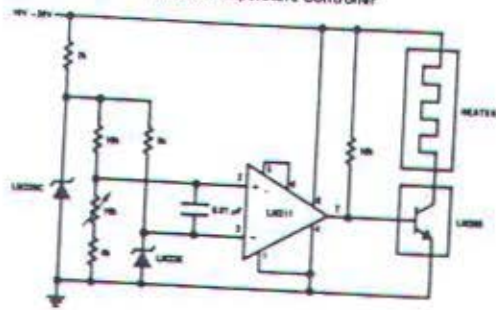


Wide Operating Supply

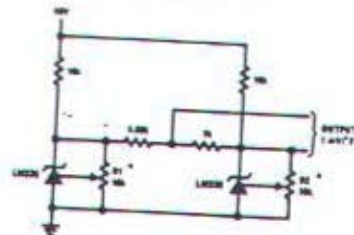


LM135/LM235/LM335, PRECISION TEMPERATURE SENSORS

Simple Temperature Controller



Fahrenheit Thermometer



* To calibrate adjust R2 for 2.564V across LM335.
Adjust R1 for correct output.

General Description

The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in °Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of $\pm 1/2^\circ\text{C}$ at room temperature and $\pm 3/4^\circ\text{C}$ over a full -55 to $+150^\circ\text{C}$ temperature range. Low cost is assured by trimming and calibration at the wafer level. The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only $60\text{ }\mu\text{A}$ from its supply, it has very low self-heating, less than 0.1°C in still air. The LM35 is rated to operate over a -55 to $+150^\circ\text{C}$ temperature range, while the LM35C is rated for a -40 to $+110^\circ\text{C}$ range (-10 with improved accuracy). The LM35 series is

available packaged in hermetic TO-46 transistor packages, while the LM35C is also available in the plastic TO-92 transistor package.

Features

- Calibrated directly in ° Celsius (Centigrade)
- Linear $+10.0\text{ mV}/^\circ\text{C}$ scale factor
- 0.5°C accuracy guaranteeable (at $+25^\circ\text{C}$)
- Rated for full -55 to $+150^\circ\text{C}$ range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts
- Less than $60\text{ }\mu\text{A}$ current drain
- Low self-heating, 0.06°C in still air
- Nonlinearity only $\pm 1/2^\circ\text{C}$ typical
- Low impedance output, $0.1\text{ }\Omega$ for 1 mA load

Connection Diagrams

TO-46
Metal Can Package*

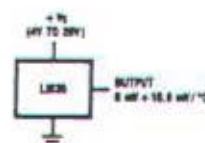


*Case is connected to negative pin

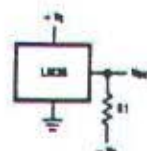
TO-92
Plastic Package



Typical Applications



Basic Centigrade Temperature Sensor ($+2^\circ\text{C}$ to $+150^\circ\text{C}$)

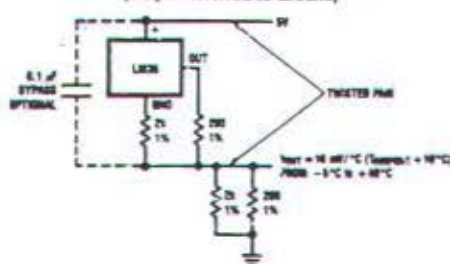


Full-Range Centigrade Temperature Sensor

Choose $R_1 = -V_S/60\text{ }\mu\text{A}$

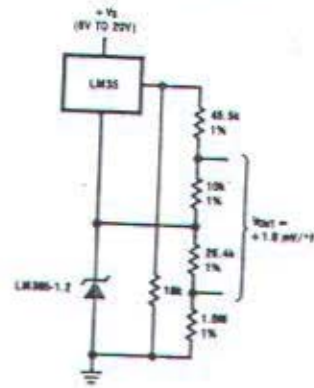
$V_{OUT} = +1,500\text{ mV at }+150^\circ\text{C}$
 $= +250\text{ mV at }+25^\circ\text{C}$
 $= -500\text{ mV at }-55^\circ\text{C}$

Two-Wire Remote Temperature Sensor
(Output Referred to Ground)



LM35 PRECISION CENTIGRADE TEMPERATURE SENSOR

Fahrenheit Thermometer



USULAN TUGAS AKHIR

PROGRAM DIII TEKNIK KOMPUTER KONTROL
FAKULTAS TEKNOLOGI INDUSTRI ITS SURABAYA

EL 4704 TUGAS AKHIR – 4 SKS

NAMA / NRP : RASTONO HENDRIYANTO
2295.030.025
MOCHAMAD ISLAN
2295.030.045
TUGAS DIBERIKAN : APRIL 1998
TUGAS DISELESAIKAN : OKTOBER 1998
DOSEN PEMBIMBING : Ir. HENDRA KUSUMA
JUDUL TUGAS AKHIR :

“ PERENCANAAN DAN PEMBUATAN SISTEM AKUSISI DATA
BERBASIS IBM PC “


URAIAN TUGAS AKHIR :

Pemanfaatan dan pengembangan teknologi pada saat ini sangatlah diperlukan, terutama dengan tujuan mempermudah dan mempercepat suatu proses pengiriman data ke dalam komputer.

Sistem akusisi data adalah suatu sistem yang digunakan untuk pengambilan (sample) data dari transducer atau sensor dari jarak jauh yang dikirimkan ke komputer. Dimana dalam hal ini sample yang dilakukan adalah pengukuran tiga parameter data. Hasil pengambilan data tersebut, dengan interface akan dihubungkan ke komputer untuk menampilkan data-data yang sudah diambil.

Dengan mata kuliah yang menunjang diharapkan tugas akhir tersebut dapat dilaksanakan dan diselesaikan dengan baik.

Mengetahui,
KETUA PROGRAM STUDI
D3 KOMPUTER KONTROL


Ir. H. M. Djoko Santoso
NIP : 130.524.518



Surabaya, April 1998
Menyetujui,
DOSEN PEMBIMBING



Ir. Hendra Kusuma
NIP : 131.846.104

USULAN TUGAS AKHIR

1. JUDUL TUGAS AKHIR :

“ PERENCANAAN DAN PEMBUATAN SISTEM AKUSISI DATA
BERBASIS IBM-PC “

2. RUANG LINGKUP :

- Elektronika Industri
- Teknik Interfacing
- Rangkaian Elektronika
- Rangkaian Listrik
- Bahasa Pemrograman Komputer

3. LATAR BELAKANG :

- Kecepatan, efisiensi dan efektifitas sangat diperlukan dalam hal pengiriman data pengukuran ke komputer.
- Diperlukan hardware yang dapat melakukan proses akusisi data.

4. PENELAAHAN STUDI :

- Mempelajari bidang studi yang berhubungan dengan proses pembuatan hardware yang berhubungan dengan judul tugas akhir.
- Mempelajari cara-cara pengimplementasian hardware dengan media komputer yang digunakan.

4. TUJUAN :

Merencanakan dan membuat hardware yang dapat melakukan proses akusisi data dengan menggunakan IBM PC.

5. RELEVANSI :

Relevansi dari hasil tugas akhir ini nantinya diharapkan dapat berguna bagi industri dan aplikasi lainnya.

6. LANGKAH-LANGKAH :

- Studi Literatur.
- Perencanaan dan pembuatan Hardware.
- Pengujian Hardware.
- Penulisan Tugas Akhir.

4. JADWAL KEGIATAN :

| No. | Kegiatan | Bulan | | | | | |
|-----|------------------------------------|-------|----|-----|----|---|----|
| | | I | II | III | IV | V | VI |
| 1. | Studi Literatur | | | | | | |
| 2. | Perencanaan dan Pembuatan Hardware | | | | | | |
| 3. | Pengujian Hardware | | | | | | |
| 4. | Penulisan Tugas Akhir | | | | | | |

RIWAYAT HIDUP



Rastono Hendriyanto, dilahirkan di Surabaya pada tanggal 26 September 1976, putra kedua (dari empat bersaudara) dari Bapak Eka Kusumawardana dan Ibu Mintaningsih. Bertempat tinggal di Wisma Permai Tengah IX/JJ 41 Surabaya.

Riwayat pendidikan

- Th. 1989 : Lulus SDN Kertajaya XII (Pucang Jajar I) Surabaya
- Th. 1992 : Lulus SMPN VI Surabaya
- Th. 1995 : Lulus SMAN IX Surabaya
- Th. 1995 : Diterima di D3 Teknik Elektro Bidang Studi Komputer Kontrol
- Fakultas Teknologi Industri - ITS

Organisasi dan kegiatan akademik yang pernah diikuti selama menjadi mahasiswa D3 Teknik Elektro Bidang Studi Komputer Kontrol FTI - ITS :

1. Menjadi Pengurus HIMA Komputer Kontrol di Bidang Kemahasiswaan pada periode 1995/1996.
2. Menjadi Pengurus HIMA Komputer Kontrol di Bidang Pendidikan pada periode 1996/1997.
3. Menjadi asisten Bahasa Pemrograman, Rangkaian Listrik, dan PLC (Programmable Logic Controller).

RIWAYAT HIDUP



Nama : Mochamad Islan, dilahirkan pada tanggal 13 Januari 1976 dari pasangan Bapak Abdul Manap (Alm.) dan Ibu Sulasanik (Alm.). Yang beralamatkan di Gunung Sari Indah Blok KK-20 Surabaya.

Riwayat Pendidikan:

1. SDN Krembangan Selatan IV No. 15 Surabaya (1982 – 1988)
2. SMPN 2 Surabaya (1988 – 1991)
3. SMAN 21 Surabaya (1991 – 1995)
4. D3 Komputer Kontrol ITS (1995 – Sekarang)

Pengalaman Akademis :

1. Asisten Praktikum Rangkaian Logika Tahun 1996
2. Asisten Praktikum Rangkaian Listrik Tahun 1997
3. Asisten Praktikum Programable Logic Control (PLC) Tahun 1997

Pengalaman Organisasi:

1. Ka-Sie Bidang Kaderisasi Himpunan Mahasiswa Komputer Kontrol ITS
Periode 1995 - 1996